

Docket No.: 67471-020

**PATENT**

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

In re Application of :  
:   
Kunihiko HAYASHI :  
:   
Serial No.: : Group Art Unit:  
:   
Filed: July 17, 2003 : Examiner:  
:   
For: PROGRAM EXECUTION APPARATUS

**CLAIM OF PRIORITY AND  
TRANSMITTAL OF CERTIFIED PRIORITY DOCUMENT**

Mail Stop Patent Application  
Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

Sir:

In accordance with the provisions of 35 U.S.C. 119, Applicant hereby claims the priority of:

**Japanese Patent Application No. 2002-218869, filed July 26, 2002,**

cited in the Declaration of the present application. A certified copy is submitted herewith.

Respectfully submitted,

MCDERMOTT, WILL & EMERY

  
Michael E. Fogarty  
Registration No. 36,139

600 13<sup>th</sup> Street, N.W.  
Washington, DC 20005-3096  
(202) 756-8000 MEF:km  
Facsimile: (202) 756-8087  
**Date: July 17, 2003**

日 本 国 特 許 庁  
JAPAN PATENT OFFICE

67471-020  
Kunihiko Hayashi  
July 17, 2003  
McDermott, Will & Emery

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日  
Date of Application:

2002年 7月26日

出 願 番 号  
Application Number:

特願2002-218869

[ST.10/C]:

[JP2002-218869]

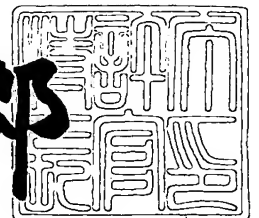
出 願 人  
Applicant(s):

松下電器産業株式会社

2003年 2月 7日

特 許 庁 長 官  
Commissioner,  
Japan Patent Office

太田信一郎



出証番号 出証特2003-3005160

【書類名】 特許願

【整理番号】 5037730150

【提出日】 平成14年 7月26日

【あて先】 特許庁長官 殿

【国際特許分類】 G06F 9/46

【発明者】

【住所又は居所】 大阪府門真市大字門真 1 0 0 6 番地 松下電器産業株式会社内

【氏名】 林 邦彦

【特許出願人】

【識別番号】 000005821

【氏名又は名称】 松下電器産業株式会社

【代理人】

【識別番号】 100090446

【弁理士】

【氏名又は名称】 中島 司朗

【手数料の表示】

【予納台帳番号】 014823

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9003742

【ブルーフの要否】 要

【書類名】 明細書

【発明の名称】 プログラム実行装置

【特許請求の範囲】

【請求項 1】 それぞれ定められた完了目標時刻以前に完了すべき複数のタスクの実行順序を決定し、各タスクを実行するプログラム実行装置であって、

各タスクには、階層関係を有する複数の種類の優先度が割り当てられており、

すでに生成されているタスク毎に、当該タスクに割り当てられた複数の種類の優先度に基づいて定まるメモリ空間上の位置において当該タスクを識別する識別子を記憶している記憶手段と、

新たに生成されたタスクに割り当てられた複数の種類の優先度、及び当該タスクを識別する識別子を受け付ける受付手段と、

前記記憶手段において、前記受付けた複数の種類の優先度に基づいて定まるメモリ空間上の位置に前記受付けた識別子を書き込む書込手段とを備え、

複数の種類の優先度に基づいて前記記憶手段において各識別子が配列されているメモリ空間上の位置の順序に従って、各識別子が識別するタスクの実行順序を決定する

ことを特徴とするプログラム実行装置。

【請求項 2】 各タスクには、前記複数の種類の優先度として、当該タスクの完了目標時刻を示す第 1 優先度と、前記第 1 優先度より下位の優先度である第 2 優先度とが割り当てられており、

前記記憶手段は、第 1 優先度として同じ完了目標時刻が割り当てられたタスクが同じグループに属するように、前記複数のタスクを分類して生成した複数のグループを有し、各グループは完了目標時刻の順に従ってメモリ空間上の定まるメモリ空間上の位置に配列されており、各グループにはタスクを識別する識別子が当該タスクに割り当てられた第 2 優先度に従って定まるメモリ空間上の位置に配列されており、

前記受付手段は、新たに生成されたタスクに割り当てられた第 1 優先度としての完了目標時刻、当該タスクに割り当てられた第 2 優先度、及び当該タスクを識別する識別子を受け付け、

前記書込手段は、前記受付けた第 1 優先度と同じ優先度を持つ識別子が含まれるグループにおいて、前記受付けた第 2 優先度に基づいて定まるメモリ空間上の位置に前記受付けた識別子を書き込み、

前記プログラム実行装置は、前記記憶手段において、第 1 優先度及び第 2 優先度に基づいて各識別子が配列されているメモリ空間上の位置の順序に従って、各識別子が識別するタスクの実行順序を決定する

ことを特徴とする請求項 1 に記載のプログラム実行装置。

【請求項 3】 前記記憶手段は、直近の完了目標時刻である第 1 優先度に基づいてタスクが分類されたグループである先頭グループに対応付けて、前記完了目標時刻である第 1 優先度を記憶しており、

前記先頭グループに後続して配列されたグループである後続グループ毎に、当該後続グループに含まれるタスクに割り当てられた完了目標時刻と、当該後続グループに先行して配列されているグループに含まれるタスクに割り当てられた完了目標時刻との時間差を、当該後続グループに対応づけて記憶しており、

前記書込手段は、前記受付手段により受付けた完了目標時刻と、前記直近の完了目標時刻とを比較して、一致する場合、前記先頭グループにおいて前記受付けた第 2 優先度に基づいて定まるメモリ空間上の位置に前記受付けた識別子を書き込み、

一致しない場合、前記後続グループ毎に、当該後続グループに対応づけられた時間差と、当該後続グループに先行して配列されているグループに含まれるタスクに割り当てられた完了目標時刻とを加算して得られる時刻と前記受付手段により受付けた完了目標時刻とを比較し、一致する場合に、当該後続グループにおいて前記受付けた第 2 優先度に基づいて定まるメモリ空間上の位置に前記受付けた識別子を書き込む

ことを特徴とする請求項 2 に記載のプログラム実行装置。

【請求項 4】 第 1 優先度が取り得る値のうちの最も低い値は、完了目標時刻を構成する所定ビット列が取り得る最大値であり、

前記記憶手段は、前記所定ビット列の最大値を有する完了目標時刻が割り当てられたタスクを含むグループを、前記完了目標時刻の順に従ってメモリ空間上の

定まる最後尾の位置において記憶しており、

前記受付手段は、新たに生成されたタスクに割り当てられた前記所定ビット列の最大値を有する完了目標時刻、当該タスクに割り当てられた第2優先度、及び当該タスクを識別する識別子を受け付け、

前記書込手段は、前記最後尾に位置するグループにおいて前記受付けた第2優先度に基づいて定まるメモリ空間上の位置に前記受付けた識別子を書き込む

ことを特徴とする請求項2に記載のプログラム実行装置。

【請求項5】 前記記憶手段は、さらに、

各グループについて、当該グループに含まれる全てのタスクの実行に要すると予測される総実行時間を、当該グループに対応づけて記憶しており、

前記受付手段は、新たに生成されたタスクの実行に要すると予測される実行時間を受け付け、

前記プログラム実行装置は、さらに、

現在時刻に、前記総実行時間と、前記実行時間とを加算して得られた時刻が、前記受付けた識別子により識別されるタスクに割り当てられた完了目標時刻より前であるか否かを判定し、前記時刻は前記完了目標時刻以降の時刻である場合に、前記受付けた識別子により識別されるタスクの実行を拒否する拒否信号を出力する判定手段を備え、

前記判定手段により前記時刻は前記完了目標時刻より前の時刻であると判定される場合、前記書込手段は、前記受付けた第1優先度と同じ優先度を持つ識別子が含まれるグループにおいて、前記受付けた第2優先度に基づいて定まるメモリ空間上の位置に前記受付けた識別子を書き込む

ことを特徴とする請求項2に記載のプログラム実行装置。

【請求項6】 前記判定手段は、前記グループに後続する全ての後続グループから当該後続グループを選択し、選択した当該後続グループについて、現在時刻に、前記当該後続グループに対応付けられた総実行時間と、前記受付けた実行時間とを加算して得られた時刻が、前記当該後続グループに含まれるタスクに割り当てられた完了目標時刻より前であるか否かを判定し、前記の選択と判定とを全ての後続グループについて繰り返し、

前記判定手段による全ての判定において前記時刻は前記完了目標時刻より前であると判定される場合、前記書込手段は、前記受付けた第 1 優先度と同じ優先度を持つ識別子が含まれるグループにおいて、前記受付けた第 2 優先度に基づいて定まるメモリ空間上の位置に前記受付けた識別子を書き込み、

前記判定手段による全ての判定のうち少なくとも 1 回前記時刻は前記完了目標時刻以降の時刻である判定される場合に、前記判定手段は前記受付けた識別子により識別されるタスクの実行を拒否する拒否信号を出力する

ことを特徴とする請求項 5 に記載のプログラム実行装置。

【請求項 7】 各タスクには、当該タスクに定められた完了目標時刻のうち所定の単位時間の整数倍からなる第 1 優先度と、前記完了目標時刻のうち前記所定の単位時間未満の部分からなる第 2 優先度とが割り当てられており、

前記記憶手段は、同じ第 1 優先度が割り当てられたタスクが同じグループに属するように、前記複数のタスクを分類して生成した複数のグループを有し、各グループは第 1 優先度に従って定まるメモリ空間上の位置に配列されており、各グループにはタスクを識別する識別子が当該タスクに割り当てられた第 2 優先度に従って定まるメモリ空間上の位置に配列されており、

前記受付手段は、新たに生成されたタスクに割り当てられた第 1 優先度、第 2 優先度、及び当該タスクを識別する識別子を受け付け、

前記書込手段は、前記受付けた第 1 優先度と同じ優先度を持つ識別子が含まれるグループにおいて、前記受付けた第 2 優先度に基づいて定まるメモリ空間上の位置に前記受付けた識別子を書き込み、

前記プログラム実行装置は、前記記憶手段において、第 1 優先度及び第 2 優先度に基づいて各識別子が配列されているメモリ空間上の位置の順序に従って、各識別子が識別するタスクの実行順序を決定する

ことを特徴とする請求項 1 に記載のプログラム実行装置。

【請求項 8】 それぞれ定められた完了目標時刻以前に完了すべき複数のタスクの実行順序を決定し、各タスクを実行するプログラム実行装置で用いられるタスク管理方法であって、

各タスクには、階層関係を有する複数の種類の優先度が割り当てられており、

前記プログラム実行装置は、すでに生成されているタスク毎に、当該タスクに割り当てられた複数の種類の優先度に基づいて定まるメモリ空間上の位置において当該タスクを識別する識別子を記憶している記憶手段を備え、

前記タスク管理方法は、

新たに生成されたタスクに割り当てられた複数の種類の優先度、及び当該タスクを識別する識別子を受け付ける受付ステップと、

前記記憶手段において、前記受付手段により受付けた複数の種類の優先度に基づいて定まるメモリ空間上の位置に前記受付けた識別子を書き込む書込ステップとを含み、

前記プログラム実行装置は、複数の種類の優先度に基づいて前記記憶手段において各識別子が配列されているメモリ空間上の位置の順序に従って、各識別子が識別するタスクの実行順序を決定する

ことを特徴とするタスク管理方法。

【請求項 9】 それぞれ定められた完了目標時刻以前に完了すべき複数のタスクの実行順序を決定し、各タスクを実行するプログラム実行装置で用いられるタスク管理プログラムであって、

各タスクには、階層関係を有する複数の種類の優先度が割り当てられており、

前記プログラム実行装置は、すでに生成されているタスク毎に、当該タスクに割り当てられた複数の種類の優先度に基づいて定まるメモリ空間上の位置において当該タスクを識別する識別子を記憶している記憶手段を備え、

前記タスク管理プログラムは、

新たに生成されたタスクに割り当てられた複数の種類の優先度、及び当該タスクを識別する識別子を受け付ける受付ステップと、

前記記憶手段において、前記受付手段により受付けた複数の種類の優先度に基づいて定まるメモリ空間上の位置に前記受付けた識別子を書き込む書込ステップとを含み、

前記プログラム実行装置は、複数の種類の優先度に基づいて前記記憶手段において各識別子が配列されているメモリ空間上の位置の順序に従って、各識別子が識別するタスクの実行順序を決定する



ことを特徴とするタスク管理プログラム。

【請求項 1 0】 それぞれ定められた完了目標時刻以前に完了すべき複数のタスクの実行順序を決定し、各タスクを実行するプログラム実行装置で用いられるタスク管理プログラムが記録されているコンピュータ読み取り可能な記録媒体であって、

各タスクには、階層関係を有する複数の種類の優先度が割り当てられており、前記プログラム実行装置は、すでに生成されているタスク毎に、当該タスクに割り当てられた複数の種類の優先度に基づいて定まるメモリ空間上の位置において当該タスクを識別する識別子を記憶している記憶手段を備え、

前記タスク管理プログラムは、

新たに生成されたタスクに割り当てられた複数の種類の優先度、及び当該タスクを識別する識別子を受け付ける受付ステップと、

前記記憶手段において、前記受付手段により受付けた複数の種類の優先度に基づいて定まるメモリ空間上の位置に前記受付けた識別子を書き込む書込ステップとを含み、

前記プログラム実行装置は、複数の種類の優先度に基づいて前記記憶手段において各識別子が配列されているメモリ空間上の位置の順序に従って、各識別子が識別するタスクの実行順序を決定する

ことを特徴とする記録媒体。

【請求項 1 1】 それぞれ定められた完了目標時刻以前に完了すべき複数のタスクの実行順序を決定し、各タスクを実行する携帯電話機であって、

各タスクには、階層関係を有する複数の種類の優先度が割り当てられており、すでに生成されているタスク毎に、当該タスクに割り当てられた複数の種類の優先度に基づいて定まるメモリ空間上の位置において当該タスクを識別する識別子を記憶している記憶手段と、

新たに生成されたタスクに割り当てられた複数の種類の優先度、及び当該タスクを識別する識別子を受け付ける受付手段と、

前記記憶手段において、前記受付手段により受付けた複数の種類の優先度に基づいて定まるメモリ空間上の位置に前記受付けた識別子を書き込む書込手段とを

備え、

複数の種類の優先度に基づいて前記記憶手段において各識別子が配列されているメモリ空間上の位置の順序に従って、各識別子が識別するタスクの実行順序を決定する

ことを特徴とする携帯電話機。

【発明の詳細な説明】

【 0 0 0 1 】

【発明の属する技術分野】

本発明は、オペレーティングシステムにおけるタスク管理機能に関し、特にタスクの実行順序を決定する技術に関する。

【 0 0 0 2 】

【従来の技術】

オペレーティングシステムの主な機能は、ハードウェア管理、タスク管理、データ管理及び入出力管理である。なかでもタスク管理は、タスクの実行順序を管理するものであり、CPUやメモリ、入出力装置等を効率良く動作させるための重要な機能である。タスクとは、プログラムの起動とその実行、終了などの流れを一括管理する制御単位のことである。OSの管理下で動作するプログラムはタスクとして扱われ、プログラムの実行、優先処理、並行処理などはすべてタスクを単位として行われる。

【 0 0 0 3 】

タスクの実行順序を決定するアルゴリズムのひとつとして、優先度法がある。優先度法は、簡単に説明すると、タスクごとに割り当てられた優先度に基づいて、各タスクの実行順序を決定する方法である。

【 0 0 0 4 】

【発明が解決しようとする課題】

しかしながら、従来の優先度法では、同じ優先度を割り当てられた複数のタスクの実行順序は先着順に決定されるため、前記複数のタスクの実行順序を制御できないという問題がある。

本発明は、上記問題を解決するために、タスクの実行順序を適切に決定するこ

とのできるタスク管理方法、タスク管理プログラム及び前記タスク管理プログラムを備えたプログラム実行装置の提供を目的とする。

#### 【0005】

##### 【課題を解決するための手段】

上記の目的を達成するために、本発明は、それぞれ定められた完了目標時刻以前に完了すべき複数のタスクの実行順序を決定し、各タスクを実行するプログラム実行装置であって、各タスクには、階層関係を有する複数の種類の優先度が割り当てられており、すでに生成されているタスク毎に、当該タスクに割り当てられた複数の種類の優先度に基づいて定まるメモリ空間上の位置において当該タスクを識別する識別子を記憶している記憶手段と、新たに生成されたタスクに割り当てられた複数の種類の優先度、及び当該タスクを識別する識別子を受け付ける受付手段と、前記記憶手段において、前記受付けた複数の種類の優先度に基づいて定まるメモリ空間上の位置に前記受付けた識別子を書き込む書込手段とを備え、複数の種類の優先度に基づいて前記記憶手段において各識別子が配列されているメモリ空間上の位置の順序に従って、各識別子が識別するタスクの実行順序を決定することを特徴とする。

#### 【0006】

##### 【発明の実施の形態】

次に、本発明に係るタスク管理方法が適用された携帯電話機1を一例として説明する。

#### 〔1〕 携帯電話機1

##### (1) 携帯電話機1の構成

携帯電話機1は、図1に示すように、送受信回路10、画面表示回路11、音声入力回路12、音声出力回路13、ボタン入力回路14、メモリ15、I/Oブリッジ16、タイマ回路17及びCPU18より構成される。

#### 【0007】

送受信回路10、画面表示回路11、音声入力回路12、音声出力回路13、ボタン入力回路14、及びタイマ回路17は、互いにシステムバス19aを介して接続される。メモリ15とCPU18とは、互いにCPUバス19bを介して

接続される。システムバス 1 9 a と CPU バス 1 9 b とは、互いに I / O ブリッジ 1 6 を介して接続される。

【 0 0 0 8 】

送受信回路 1 0 は、e メール又は発信者電話番号等の通信データ並びに音声データの送受信を行う。受信時においては、受信した通信データはメモリ 1 5 に書き込まれると同時に、画面表示回路 1 1 によって、メモリ 1 5 から読み出され、ディスプレイに表示される。受信した音声データは、音声出力回路 1 3 を通じて発音される。

送信時においては、ボタン入力回路 1 4 を通じて入力される電話番号等を受け付け、送受信回路 1 0 を通じて発信処理する。相手先と回線が繋がった後は、音声入力回路 1 2 を通じて入力される音声を発信処理する。

【 0 0 0 9 】

メモリ 1 5 は、通信データ、画面データ、音声データ、ボタン入力データ等を記憶している。その他に、リアルタイムオペレーティングシステムと呼ばれる制御プログラムと、通信データ、画面データ、音声データ及び入力データの各データを実行するための複数個の命令プログラムとを記憶している。

タイマ回路 1 7 は、タイムアウト信号を CPU 1 8 に出力して、実行されるプログラム及びデータの切換えタイミングを通知する。

( 2 ) 携帯電話機 1 における CPU 1 8 の動作

CPU 1 8 は、メモリ 1 5 より制御プログラムを読み込み、その制御プログラムに従って、メモリ 1 5 より順に 1 個の命令プログラムを読み出す。そして、読出した命令プログラムを解読し、解読した命令を実行する。なお、CPU 1 8 は、前記命令を実行する際に、必要となる通信データ、画面データ、音声データ及び入力データの各データをメモリ 1 5 より読み出す。

【 0 0 1 0 】

更に、CPU 1 8 は、前記命令を実行して得られた演算結果をメモリ 1 5 に書き込む。演算結果は、携帯電話機においては、上記した通信データ、画面データ、音声データが該当し、これらがメモリから読み出され、送信、再生処理される。

本実施の形態では、以下、CPU 1 8 が命令プログラムを実行する実行単位

をタスクと呼ぶ。また、制御プログラムに従って、各命令プログラムの実行順序を決めることをスケジューリングと呼ぶ。

【 0 0 1 1 】

タスクのスケジューリング方法について、以下に説明する。

〔 2 〕 タスクのスケジューリング方法

タスクのスケジューリングは、図 2 に示すように、タスクスケジューリング部 2 0、プログラム記憶部 2 1、タイマ制御部 2 2 及び実行制御部 2 3 より構成される。

【 0 0 1 2 】

このうち、プログラム記憶部 2 1、タイマ制御部 2 2 及び実行制御部 2 3 に関しては、従来のタスクスケジューリングの技術として公知であるので説明を省略し、タスクスケジューリング部 2 0 について、説明する。

タスクスケジューリング部 2 0 は、タスク受付部 2 0 1、スケジューリング可能性判定部 2 0 2、スケジューリング部 2 0 3、タスク情報記憶部 2 0 4 及びタスク切換部 2 0 5 を有している。

【 0 0 1 3 】

( 1 ) タスク受付部 2 0 1

タスク受付部 2 0 1 は、ユーザからの操作等により発生したタスクを受け付ける。具体的には、実行制御部 2 3 よりタスクの処理要求信号を受け付ける。この後、プログラム記憶部 2 1 より当該タスクの「タスク情報」、「優先度」、「デッドライン時刻」（以降、DL 時刻と呼ぶ）及び「最悪実行時間」を読み出す。

「タスク情報」は、プログラム開始アドレスとスタックポインタとからなる。プログラム開始アドレスは、タスクが書き込まれたアドレスである。スタックポインタは、タスクの状態が一時的にスタックに退避される場合に、タスクの状態が書き込まれる位置情報である。

【 0 0 1 4 】

「優先度」は、タスクの実行順序を決定するための基準を数値で示したものである。本実施の形態では、優先度を“高”、“中”及び“低”の 3 レベルとする

。

「DL時刻」は、タスクの実行を完了すべき目標時刻であり、“時／分／秒／秒以下 (msec,  $\mu$  sec)” で表記する。なお、本実施の形態では、「DL時刻」は前記優先度より上位の優先度であるとする。

【0015】

「最悪実行時間」は、タスクの実行開始から実行終了までに要すると予測される時間である。

タスク受付部 201 は、これら読み出した「タスク情報」、「優先度」、「DL時刻」、及び「最悪実行時間」を、スケジューリング可能性判定部 202 へ出力する。

(2) タスク情報記憶部 204

タスク情報記憶部 204 は、各タスクに割り当てられた「DL時刻」、及び「優先度」を管理するために設けられており、図 3 に示すように、先頭ポインタ 30、グループ 400、グループ 500、・・・を有している。

【0016】

先頭ポインタ 30、グループ 400、グループ 500、・・・は、先頭ポインタ 30 を先頭とし、グループ 400、グループ 500、・・・の順に配列を構成している。

グループ 400 は、同一 DL 時刻が割り当てられた複数のタスクに関する情報をまとめて管理するためのものであり、DCB 40、TCB 41、TCB 42、・・・からなる。

【0017】

DCB 40、TCB 41、TCB 42、・・・は、DCB 40 を先頭とし、TCB 41、TCB 42、・・・の順に配列を構成している。TCB (タスクコントロールブロック: Task Control Block) は、タスクを管理するための管理情報である。1 個の TCB が、1 タスクに対応する。DCB (デッドラインコントロールブロック: Deadline Control Block) は、各グループに含まれる TCB が管理するタスクに割り当てられた DL 時刻を管理するための管理情報である。

【0018】

先頭ポインタ 3 0 は、DL 基準時刻 3 0 1 及び DCB リンクアドレス 3 0 2 を含んでいる。DL 基準時刻 3 0 1 は、各タスクに割り当てられた全ての DL 時刻のうち最も早い時刻である。DCB リンクアドレス 3 0 2 は、グループ 4 0 0 に含まれる DCB 4 0 が書き込まれた位置を示すアドレスである。

DCB 4 0 は、相対 DL 値 4 0 1、最悪実行時間 4 0 2、DCB リンクアドレス 4 0 3 及び TCB リンクアドレス 4 0 4 を含んでいる。相対 DL 値 4 0 1 は、DL 基準時刻 3 0 1 から、グループ 4 0 0 に含まれる TCB が管理するタスクに割り当てられた DL 時刻までのあいだの時間である。

#### 【 0 0 1 9 】

最悪実行時間 4 0 2 は、グループ 4 0 0 に含まれる全ての TCB が管理するタスクに割り当てられた最悪実行時間を加算して得られる合計時間である。DCB リンクアドレス 4 0 3 は、グループ 5 0 0 に含まれる DCB 5 0 が書き込まれた位置を示すアドレスである。TCB リンクアドレス 4 0 4 は、TCB 4 1 が書き込まれた位置を示すアドレスである。

#### 【 0 0 2 0 】

TCB 4 1 は、優先度 4 1 1、タスク情報 4 1 2 及び TCB リンクアドレス 4 1 3 を含んでいる。優先度 4 1 1 は、TCB 4 1 が管理するタスクに割り当てられた優先度の値である。タスク情報 4 1 2 は、TCB 4 1 により管理されるタスクが書き込まれた位置を示すアドレスである。TCB リンクアドレス 4 1 3 は、TCB 4 2 が書き込まれた位置を示すアドレスである。

#### 【 0 0 2 1 】

TCB 4 2 は、優先度 4 2 1、タスク情報 4 2 2 及び TCB リンクアドレス 4 2 3 を含んでいる。優先度 4 2 1 は上記優先度 4 1 1 と同様の構成であり、タスク情報 4 2 2 は上記タスク情報 4 1 2 と同様の構成であり、さらに TCB リンクアドレス 4 2 3 は上記 TCB リンクアドレス 4 1 3 と同様の構成であるので、説明を省略する。

#### 【 0 0 2 2 】

グループ 5 0 0 は、DCB 5 0、TCB 5 1、・・・を含んでいる。

DCB 5 0 は、相対 DL 値 5 0 1、最悪実行時間 5 0 2、DCB リンクアドレ

ス 5 0 3 及び T C B リンクアドレス 5 0 4 を含んでいる。相対 D L 値 5 0 1 は上記相対 D L 値 4 0 1 と同様の構成であり、最悪実行時間 5 0 2 は上記最悪実行時間 4 0 2 と同様の構成であり、D C B リンクアドレス 5 0 3 は上記 D C B リンクアドレス 4 0 3 と同様の構成であり、さらに T C B リンクアドレス 5 0 4 は上記 T C B リンクアドレス 4 0 4 と同様の構成であるので、説明を省略する。

## 【 0 0 2 3 】

T C B 5 1 は、優先度 5 1 1、タスク情報 5 1 2 及び T C B リンクアドレス 5 1 3 を含んでいる。優先度 5 1 1 は上記優先度 4 1 1 と同様の構成であり、タスク情報 5 1 2 は上記タスク情報 4 1 2 と同様の構成であり、さらに T C B リンクアドレス 5 1 3 は上記 T C B リンクアドレス 4 1 3 と同様の構成であるので、説明を省略する。

## 【 0 0 2 4 】

なお、先頭ポインタ 3 0、グループ 4 0 0、グループ 5 0 0、・・・による配列は、各グループに格納されている T C B に割り振られた D L 時刻順に構成される。例えば、タスク情報記憶部 2 0 4 は、T C B 4 1 が含まれるグループ 4 0 0 の相対 D L 時間 4 0 1 に D L 時刻 “ 6 / 35 / 2 7 ” の値と D L 基準時刻との差を記憶しており、T C B 5 1 が含まれるグループ 5 0 0 の相対 D L 時間 5 0 1 に D L 時刻 “ 6 / 35 / 4 3 ” の値をグループ 4 0 0 の D L 時刻との差を記憶している。

## 【 0 0 2 5 】

また、T C B 4 1、T C B 4 2、・・・による配列は、各 T C B に割り振られた優先度の値の高いランク順に構成される。例えば、タスク情報記憶部 2 0 4 は、優先度 4 1 1 の領域に “ 高 ” を記憶しており、優先度 4 2 1 の領域に “ 中 ” を記憶している。

さらに、タスク情報記憶部 2 0 4 は、最も遅い D L 時刻を割り当てられた T C B を配列した D C B に含まれる D C B リンクアドレスの領域に、最終 D C B を指定するアドレスを記憶している。タスク情報記憶部 2 0 4 は、前記最終 D C B 内に、D L 時刻のデータ値が “ F F / F F / F F ” である複数の T C B を、割り当てられた第 2 優先度に基づいて配列して記憶している。“ F F / F F / F F ” は、D L 時刻を構成する所定のビット列が取り得る最大値とする。つまり、T C B



のDL時刻に“FF/FF/FF”を割り当てることにより、前記TCBの優先度を最低に設定することが可能となる。

### (3) スケジューリング可能性判定部 2 0 2

スケジューリング可能性判定部 2 0 2 は、前記当該タスクがスケジューリングされた場合、スケジューリングされている全てのタスクの実行をDL時刻より以前に完了することが出来るか否かを判定するために設けられている。

#### 【0 0 2 6】

スケジューリング可能性判定部 2 0 2 は、 $i$ 、 $X$ 及び $Y$ という変数を有している。また、スケジューリング可能性判定部 2 0 2 は、タスク受付部 2 0 1 より当該タスクに割り当てられたタスク情報、優先度、DL時刻及び最悪実行時間を受け取り、タスク情報記憶部 2 0 4 より先頭ポインタに含まれるDL基準時刻を読み出し、さらに、タイマ制御部 2 2 より現在時刻を取得する。

#### 【0 0 2 7】

そして、スケジューリング可能性判定部 2 0 2 は、 $n$ 回（変数  $i = 1, 2, \dots, n$ ）、以下の処理を繰り返す。

①、変数 $X$ に前記DL基準時刻を代入し、変数 $Y$ に前記現在時刻と前記当該タスクの最悪実行時間との合計値を代入する。

②、タスク情報記憶部 2 0 4 よりDCB ( $i$ ) に含まれる相対DL値 ( $i$ ) と最悪実行時間 ( $i$ ) とを読み出す。

#### 【0 0 2 8】

③、変数 $X$ に変数 $X$ と相対DL値 ( $i$ ) との合計値を代入し、変数 $Y$ に変数 $Y$ と最悪実行時間 ( $i$ ) との合計値を代入する。

④、変数 $X$ と変数 $Y$ との比較を行う。

⑤、変数 $i$ に変数 $i$ と1との合計値を代入する。

ここまでの処理が、繰り返し処理の内容となる。

#### 【0 0 2 9】

スケジューリング可能性判定部 2 0 2 は、上記変数 $X$ と変数 $Y$ との比較を繰り返した結果において、1回でも変数 $X \leq$  変数 $Y$ であった場合、すなわちDL時刻が現在時刻に最悪実行時間を加算した時刻と同時刻又は前の時刻である場合に、

スケジューリング不可信号を実行制御部 2 3 へ出力する。

他方、上記変数 X と変数 Y との比較を繰り返した全ての結果において、変数  $X >$  変数 Y であった場合、すなわち DL 時刻が現在時刻に最悪実行時間を加算した時刻より後の時刻である場合に、当該タスクのタスク情報、優先度、DL 時刻及び最悪実行時間をスケジューリング部 2 0 3 へ出力する。

#### (4) スケジューリング部 2 0 3

スケジューリング部 2 0 3 は、当該タスクを管理する T C B を、DL 時刻と優先度とに基づいて、タスク情報記憶部 2 0 4 において定まるメモリ空間上の位置に書き込むために設けられている。

##### 【 0 0 3 0 】

スケジューリング部 2 0 3 は、i 及び T という変数を有しており、スケジューリング可能性判定部 2 0 2 より当該タスクのタスク情報、優先度、DL 時刻及び最悪実行時間を受け取り、タスク情報記憶部 2 0 4 より先頭ポインタに含まれる D C B リンクアドレスと DL 基準時刻とを読み出す。

そして、スケジューリング部 2 0 3 は、変数 i に 1 を代入し、変数 T に前記 DL 基準時刻を代入して、記憶部 2 0 4 における D C B ( 1 ) のデータの有無を判別する。判別結果によって以下のように各処理を行なう。

##### 【 0 0 3 1 】

①、 D C B ( 1 ) のデータが無いと判定する場合、記憶部 2 0 4 への書き込みを行う。(処理 A とする)

②、 一方、 D C B ( 1 ) のデータが有ると判定する場合、当該タスクの DL 時刻と変数 T とを比較する。

②- 1、 タスクの DL 時刻 < 変数 T の場合に、記憶部 2 0 4 への書き込みを行う。(処理 B とする)

②- 2、 タスクの DL 時刻 = 変数 T の場合に、記憶部 2 0 4 への書き込みを行う。(処理 E とする)

②- 3、 さらに、タスクの DL 時刻 > 変数 T の場合に、変数 i に 1 を加算した後、 D C B ( i ) データの有無を判定する。

##### 【 0 0 3 2 】

②-3-1、 D C B ( i ) データが無いと判定する場合、記憶部 2 0 4 への書き込みを行う。(処理 C とする)

②-3-2、 他方、D C B ( i ) データは有ると判定する場合、D C B ( i ) に含まれる相対 D L 値を読み出す。さらに、変数 T に変数 T と前記相対 D L 値との合計値を代入し、当該タスクの D L 時刻と変数 T とを比較する。

【 0 0 3 3 】

②-3-2-1、 タスクの D L 時刻 < 変数 T の場合に、記憶部 2 0 4 への書き込みを行う。(処理 D とする)

②-3-2-2、 タスクの D L 時刻 = 変数 T の場合に、記憶部 2 0 4 への書き込みを行う。(処理 E とする)

以下に、処理 A ~ E を更に詳細に説明する。

[処理 A の詳細]

スケジューリング部 2 0 3 は、以下の処理を行なう。

タスク情報記憶部 2 0 4 に T C B ( a ) の領域を生成して、書き込む。即ち、タスク情報記憶部 2 0 4 における T C B ( a ) の優先度の領域に、スケジューリング可能性判定部 2 0 2 より受取った当該タスクの優先度を書き込み、T C B ( a ) のタスク情報の領域に、当該タスクのタスク情報を書き込み、さらに、T C B ( a ) の T C B リンクアドレスの領域に、N U L L の値を代入する。

ついで、タスク情報記憶部 2 0 4 に、D C B ( a ) の領域を生成して、書き込む。即ち、タスク情報記憶部 2 0 4 における D C B ( a ) の相対 D L 値の領域に、0 の値を代入し、D C B ( a ) の最悪実行時間の領域に、当該タスクの最悪実行時間を書き込み、D C B ( a ) の D C B リンクアドレスの領域に、N U L L の値を代入し、D C B ( a ) の T C B リンクアドレスの領域に、T C B ( a ) の位置を示すアドレスを書き込む。なお、この処理により、D C B ( a ) と T C B ( a ) とは、タスク情報記憶部 2 0 4 において配列を構成する。

【 0 0 3 4 】

また、スケジューリング部 2 0 3 は、タスク情報記憶部 2 0 4 における D L 基準時刻 3 0 1 の領域に、当該タスクの D L 時刻を上書きする。

さらに、スケジューリング部 2 0 3 は、タスク情報記憶部 2 0 4 における D C

Bリンクアドレス302の領域に、DCB(a)の位置を示すアドレスを上書きする。なお、この処理により、先頭ポインタとDCB(a)とは、タスク情報記憶部204において配列を構成する。

## 【0035】

スケジューリング部203は、タイマ制御部22に当該タスクのDL時刻を出力する。

## [処理Bの詳細]

スケジューリング部203は、タスク情報記憶部204にTCB(b)の領域を生成して、書き込む。即ち、TCB(b)の優先度の領域に、スケジューリング可能性判定部202より受取った当該タスクの優先度を書き込み、TCB(b)のタスク情報の領域に、当該タスクのタスク情報を書き込み、TCB(b)のTCBリンクアドレスの領域に、NULLの値を代入する。

## 【0036】

また、スケジューリング部203は、タスク情報記憶部204にDCB(b)の領域を生成して、書き込む。即ち、DCB(b)の相対DL値の領域に、0の値を代入し、DCB(b)の最悪実行時間の領域に、当該タスクの最悪実行時間を書き込み、DCB(b)のDCBリンクアドレスの領域に、DCB(1)の位置を示すアドレスを書き込む。なお、この処理により、DCB(b)とDCB(1)とは、タスク情報記憶部204において配列を構成する。

## 【0037】

スケジューリング部203は、さらに、DCB(b)のTCBリンクアドレスの領域に、TCB(b)の位置を示すアドレスを書き込む。この処理により、DCB(b)とTCB(b)とは、タスク情報記憶部204において配列を構成する。

また、スケジューリング部203は、タスク情報記憶部204におけるDCB(1)の相対DL値の領域に、(T-当該タスクのDL時刻)の値を上書きし、DL基準時刻301の領域に、当該タスクのDL時刻を上書きし、さらに、DCBリンクアドレス302の領域に、DCB(b)の位置を示すアドレスを上書きする。なお、この処理により、先頭ポインタとDCB(b)とは、タスク情報記

憶部 2 0 4 において配列を構成する。

【 0 0 3 8 】

スケジューリング部 2 0 3 は、タイマ制御部 2 2 に当該タスクの D L 時刻を出力する。

[処理 C の詳細]

スケジューリング部 2 0 3 は、タスク情報記憶部 2 0 4 に T C B ( c ) の領域を生成して、書き込む。即ち、タスク情報記憶部 2 0 4 における T C B ( c ) の優先度の領域に、スケジューリング可能性判定部 2 0 2 より受取った当該タスクの優先度を書き込み、T C B ( c ) のタスク情報の領域に、当該タスクのタスク情報を書き込み、T C B ( c ) の T C B リンクアドレスの領域に、N U L L の値を代入する。

【 0 0 3 9 】

また、スケジューリング部 2 0 3 は、タスク情報記憶部 2 0 4 に D C B ( c ) の領域を生成して、書き込む。即ち、D C B ( c ) の相対 D L 値の領域に、(当該タスクの D L 時刻 - T ) の値を代入し、D C B ( c ) の最悪実行時間の領域に、当該タスクの最悪実行時間を書き込み、D C B ( c ) の D C B リンクアドレスの領域に、N U L L の値を代入し、さらに、D C B ( c ) の T C B リンクアドレスの領域に、T C B ( c ) の位置を示すアドレスを書き込む。

【 0 0 4 0 】

なお、この処理により、D C B ( c ) と T C B ( c ) とは、タスク情報記憶部 2 0 4 において配列を構成する。

スケジューリング部 2 0 3 は、さらに、タスク情報記憶部 2 0 4 における D C B ( i ) の D C B リンクアドレスの領域に、D C B ( c ) の位置を示すアドレスを上書きする。この処理により、D C B ( i ) と D C B ( c ) とは、タスク情報記憶部 2 0 4 において配列を構成する。

[処理 D の詳細]

スケジューリング部 2 0 3 は、タスク情報記憶部 2 0 4 に T C B ( d ) の領域を生成して、書き込む。即ち、T C B ( d ) の優先度の領域に、スケジューリング可能性判定部 2 0 2 より受取った当該タスクの優先度を書き込み、T C B ( d

）のタスク情報の領域に、当該タスクのタスク情報を書き込み、TCB（d）のTCBリンクアドレスの領域に、NULLの値を代入する。

【0041】

スケジューリング部203は、タスク情報記憶部204にDCB（d）の領域を生成して、書き込む。即ち、DCB（d）の相対DL値の領域に、（当該タスクのDL時刻-T）の値を代入し、DCB（d）の最悪実行時間の領域に、当該タスクの最悪実行時間を書き込み、DCB（d）のDCBリンクアドレスの領域に、DCB（i）の位置を示すアドレスを書き込む。なお、この処理により、DCB（d）とDCB（i）とは、タスク情報記憶部204において配列を構成する。

【0042】

また、スケジューリング部203は、タスク情報記憶部204におけるDCB（d）のTCBリンクアドレスの領域に、TCB（d）の位置を示すアドレスを書き込む。この処理により、DCB（d）とTCB（d）とは、タスク情報記憶部204において配列を構成する。

さらに、スケジューリング部203は、タスク情報記憶部204におけるDCB（i-1）のDCBリンクアドレスの領域に、DCB（d）の位置を示すアドレスを上書きする。この処理により、DCB（i-1）とDCB（d）とは、タスク情報記憶部204において配列を構成する。

[処理Eの詳細]

スケジューリング部203は、変数jを有し、初期値を1に設定した後、タスク情報記憶部204におけるDCB（i）に含まれるTCBリンクアドレスにより指定されるTCB（j）を認識する。

【0043】

そして、タスク情報記憶部204からTCB（j）の優先度を読み出し、当該タスクの優先度値とTCB（j）の優先度とを比較する。

当該タスクの優先度がTCB（j）の優先度より低い、または等しい場合、変数jに1を加算する。そして、再び、タスク情報記憶部204にTCB（j）が記憶されているか否かを判定する。

## 【 0 0 4 4 】

T C B ( j ) が記憶されている限り、当該タスクの優先度と T C B ( j ) の優先度との比較処理、及び変数 j に 1 を加算する処理を繰り返す。

スケジューリング部 2 0 3 は、もはや、T C B ( j ) が記憶されていない場合、又は当該タスクの優先度が T C B ( j ) の優先度より高い場合、タスク情報記憶部 2 0 4 に T C B ( e ) の領域を生成して、書き込む。

## 【 0 0 4 5 】

書き込む内容は、T C B ( e ) の優先度の領域に、スケジューリング可能性判定部 2 0 2 より受取った当該タスクの優先度を、T C B ( e ) のタスク情報の領域に、当該タスクのタスク情報を、T C B ( e ) の T C B リンクアドレスの領域に、T C B ( j ) の位置を示すアドレスをそれぞれ書き込む。

なお、この処理により、T C B ( e ) と T C B ( j ) とは、タスク情報記憶部 2 0 4 において配列を構成する。

## 【 0 0 4 6 】

続いて、スケジューリング部 2 0 3 は、タスク情報記憶部 2 0 4 における T C B ( j - 1 ) の T C B リンクアドレスの領域に、T C B ( e ) の位置を示すアドレスを書き込む。この処理により、T C B ( j - 1 ) と T C B ( e ) とは、タスク情報記憶部 2 0 4 において配列を構成する。

ついで、スケジューリング部 2 0 3 は、タスク情報記憶部 2 0 4 における D C B ( i ) の最悪実行時間の領域より、最悪実行時間を読み出し、その時間と、当該タスクの最悪実行時間とを加算した合計時間を、タスク情報記憶部 2 0 4 における D C B ( i ) の最悪実行時間の領域に上書きする。

## ( 5 ) タスク切換部 2 0 5

タスク切換部 2 0 5 は、C P U 1 8 により実行されるタスクを切り換えるために設けられる。

## 【 0 0 4 7 】

タスク切換部 2 0 5 は、タイマ制御部 2 2 よりタイムアウト信号を受け取る場合、タスクの切り換えを行う。(処理 F とする)

タスク切換部 2 0 5 は、実行制御部 2 3 より終了信号を受け取る場合、タスク

情報記憶部 2 0 4 における T C B 4 1 に含まれる T C B リンクアドレス 4 1 3 を読み出す。さらに、タスク切換部 2 0 5 は、読出した T C B リンクアドレス 4 1 3 の値は N U L L 値であるか否かを判定する。

## 【 0 0 4 8 】

タスク切換部 2 0 5 は、前記 T C B リンクアドレス 4 1 3 の値が、N U L L 値である場合、タスクの切り換えを行う。（処理 F とする）

他方、前記 T C B リンクアドレス 4 1 3 の値が、N U L L 値でない場合、タスクの切り換えを行う。（処理 G とする）

以下に、処理 F 及び処理 G を詳細に説明する。

## [処理 F の詳細]

タスク切換部 2 0 5 は、タスク情報記憶部 2 0 4 より先頭ポインタ 3 0、D C B 4 0 及び D C B 5 0 を読み出し、現在の D L 基準時刻と D C B 5 0 に含まれる相対 D L 値との合計値を、タスク情報記憶部 2 0 4 における D L 基準時刻 3 0 1 の領域に上書きし、D C B 5 0 の位置を示すアドレスを、タスク情報記憶部 2 0 4 における D C B リンクアドレス 3 0 2 の領域に上書きし、さらに、D C B 5 0 に含まれる相対 D L 値 5 0 1 の領域に、0 値を上書きする。

## 【 0 0 4 9 】

また、タスク切換部 2 0 5 は、D C B 4 0 及び D C B 4 0 に接続される全ての T C B をタスク情報記憶部 2 0 4 より消去する。

続いて、タスク切換部 2 0 5 は、タスク情報記憶部 2 0 4 より D C B 5 0 に含まれる T C B リンクアドレスにより指定される T C B 5 1 を読み出し、読出した T C B 5 1 に含まれるタスク情報 5 1 2 を実行制御部 2 3 へ出力する。

## [処理 G の詳細]

タスク切換部 2 0 5 は、タスク情報記憶部 2 0 4 より D C B 4 0、T C B 4 1 及び T C B 4 2 を読み出し、D C B 4 0 に含まれる T C B リンクアドレス 4 0 4 の領域に、T C B 4 2 の位置を示すアドレスを上書きする。

## 【 0 0 5 0 】

続いて、タスク情報記憶部 2 0 4 における T C B 4 1 を消去し、T C B 4 2 に含まれるタスク情報 4 2 2 を、実行制御部 2 3 に出力する。



### 〔 3 〕 タスクスケジューリング部 2 0 の動作

タスクスケジューリング部 2 0 の動作について、図 4 に示すシーケンス図を用いて説明する。

#### 【 0 0 5 1 】

実行制御部 2 3 は、当該タスクの生成をタスク受付部 2 0 1 へ通知する（ステップ S 1 0 1）。

タスクスケジューリング部 2 0 は、プログラム記憶部 2 1 における当該タスクの位置を示すアドレスを送信し（ステップ S 1 0 2）、当該タスクのタスク情報を読み出す（ステップ S 1 0 3）。読出したタスク情報に基づいて、当該タスクのスケジューリング可能性判定を行う（ステップ S 1 0 4）。

#### 【 0 0 5 2 】

判定の結果、当該タスクのスケジューリングが不可能な場合、スケジューリング不可信号を実行制御部 2 3 へ出力する（ステップ S 1 0 5）。

判定の結果、可能な場合は、当該タスクのスケジューリングを行う（ステップ S 1 0 6）。このとき、先頭ポインタ 3 0 に含まれる D L 基準時刻 3 0 1 を変更する場合、タイマ制御部 2 2 へ D L 時刻を出力する（ステップ S 1 0 7）。

#### 【 0 0 5 3 】

タスク切換部 2 0 5 は、タイマ制御部 2 2 よりタイムアウト信号を受け取り（ステップ S 1 0 8）り、実行制御部 2 3 よりタスク終了信号を受け取る（ステップ S 1 0 9）と、タスクの切換えを行う（ステップ S 1 1 0）。

### （ 1 ） スケジューリング可能性判定の動作

スケジューリング可能性判定の動作を、図 5 に示すフローチャートを用いて説明する。

#### 【 0 0 5 4 】

スケジューリング可能性判定部 2 0 2 は、タスク受付部 2 0 1 より当該タスクのタスク情報、優先度、D L 時刻及び最悪実行時間を受け取り（ステップ S 2 0 1）、タスク情報記憶部 2 0 4 より D L 基準時刻を読み出し（ステップ S 2 0 2）、タイマ制御部 2 2 より現在時刻を取得する（ステップ S 2 0 3）。

続いて、変数 i に 1 を、変数 X に前記 D L 基準時刻を、変数 Y に前記現在時刻

と前記当該タスクの最悪実行時間との合計値をそれぞれ代入する（ステップ S 2 0 4）。

#### 【 0 0 5 5 】

そして、変数  $i$  が  $n$  になるまで、ステップ 2 0 6 から 2 1 0 までの処理を繰り返す（ステップ S 2 0 5）。

即ち、タスク情報記憶部 2 0 4 より D C B (  $i$  ) に含まれる相対 D L 値 (  $i$  ) と最悪実行時間 (  $i$  ) とを読み出し（ステップ S 2 0 6）、変数 X に変数 X と相対 D L 値 (  $i$  ) との合計値を代入し、さらに変数 Y に変数 Y と最悪実行時間 (  $i$  ) との合計値を代入する（ステップ S 2 0 7）。

#### 【 0 0 5 6 】

この状態で、変数 X と変数 Y との値を比較を行う（ステップ S 2 0 8）。

比較の結果、変数  $X \leq$  変数 Y である場合（ステップ S 2 0 8 で  $\leq$ ）、スケジューリング不可信号を実行制御部 2 3 へ出力する（ステップ S 2 0 9）が、変数  $X >$  変数 Y である場合（ステップ S 2 0 8 で  $>$ ）、変数  $i$  に変数  $i$  と 1 との合計値を代入し（ステップ S 2 1 0）、上記の処理を繰り返す（ステップ S 2 1 1）。

#### 【 0 0 5 7 】

繰り返しを終えると、タスク情報、優先度、D L 時刻及び最悪実行時間をスケジューリング部 2 0 3 へと出力する（ステップ S 2 1 2）。

### ( 2 ) スケジューリングの動作

スケジューリングの動作を、図 6 に示すフローチャートを用いて説明する。

スケジューリング部 2 0 3 は、スケジューリング可能性判定部 2 0 2 より当該タスクのタスク情報、優先度、D L 時刻及び最悪実行時間を受け取り（ステップ S 3 0 1）、タスク情報記憶部 2 0 4 より先頭ポインタに含まれる D C B リンクアドレスと D L 基準時刻とを読み出す（ステップ S 3 0 2）。

#### 【 0 0 5 8 】

そして、変数  $i$  に 1 を、変数 T に前記 D L 基準時刻をそれぞれ代入する（ステップ S 3 0 3）。

$i=1$  であるので、D C B ( 1 ) の有無を判別し（ステップ S 3 0 4）、D C B ( 1 ) が無い場合（ステップ S 3 0 4 で無）、処理 A（図 7 参照）を行い、D C B

(1) が有る場合 (ステップ S 3 0 4 で有)、当該タスクの D L 時刻と変数 T とを比較する (ステップ S 3 0 5)。

当該タスクの D L 時刻 < 変数 T である場合 (ステップ S 3 0 5 で <)、処理 B (図 8 参照) を行うし、一方、当該タスクの D L 時刻 = 変数 T である場合 (ステップ S 3 0 5 で =)、処理 E (図 1 1 参照) を行う。

#### 【 0 0 5 9 】

さらに、当該タスクの D L 時刻 > 変数 T である場合 (ステップ S 3 0 5 で >)、変数 i に変数 i と 1 との合計値を代入し (ステップ S 3 0 6)、D C B ( i ) の有無を判定する (ステップ S 3 0 7)。

D C B ( i ) が無い場合 (ステップ S 3 0 7 で無)、処理 C (図 9 参照) を行う一方、D C B ( i ) が有る場合 (ステップ S 3 0 7 で有)、D C B ( i ) に含まれる相対 D L 値を読み出し (ステップ S 3 0 8)、変数 T に変数 T と前記相対 D L 値との合計値を代入し (ステップ S 3 0 9)、当該タスクの D L 時刻と変数 T とを比較する (ステップ S 3 1 0)。

#### 【 0 0 6 0 】

比較の結果、当該タスクの D L 時刻 < 変数 T である場合 (ステップ S 3 1 0 で <)、処理 D (図 10 参照) を行う。

他方、当該タスクの D L 時刻 = 変数 T である場合 (ステップ S 3 1 0 で =)、処理 E を行う。

また、当該タスクの D L 時刻 > 変数 T である場合には (ステップ S 3 1 0 で >)、ステップ S 3 0 6 に戻り、ステップ 3 1 0 までの処理を繰り返す。

#### ① 処理 A の動作

処理 A の動作を、図 7 に示すフローチャートを用いて説明する。

#### 【 0 0 6 1 】

スケジューリング部 2 0 3 は、タスク情報記憶部 2 0 4 に T C B ( a ) の領域を生成し (ステップ S 4 0 1)、優先度の領域に当該タスクの優先度を書き込み (ステップ S 4 0 2)、タスク情報の領域に当該タスクのタスク情報を書き込み (ステップ S 4 0 3)、T C B リンクアドレスの領域に、N U L L の値を代入する (ステップ S 4 0 4)。

## 【 0 0 6 2 】

スケジューリング部 2 0 3 は、タスク情報記憶部 2 0 4 に D C B ( a ) の領域を生成して ( ステップ S 4 0 5 ) 、相対 D L 値の領域に 0 の値を代入し ( ステップ S 4 0 6 ) 、最悪実行時間の領域に当該タスクの最悪実行時間を書き込み ( ステップ S 4 0 7 ) 、 D C B リンクアドレスの領域に N U L L の値を代入し ( ステップ S 4 0 8 ) 、 T C B リンクアドレスの領域に T C B ( a ) の位置を示すアドレスを書き込む ( ステップ S 4 0 9 ) 。

## 【 0 0 6 3 】

続いて、タスク情報記憶部 2 0 4 における D L 基準時刻 3 0 1 の領域に、当該タスクの D L 時刻を上書きし ( ステップ S 4 1 0 ) 、 D C B リンクアドレス 3 0 2 の領域に D C B ( a ) の位置を示すアドレスを上書きする ( ステップ S 4 1 1 ) 。さらに、スケジューリング部 2 0 3 は、タイマ制御部 2 2 に当該タスクの D L 時刻を出力する ( ステップ S 4 1 2 ) 。

## ② 処理 B の動作

処理 B の動作を、図 8 に示すフローチャートを用いて説明する。

## 【 0 0 6 4 】

スケジューリング部 2 0 3 は、タスク情報記憶部 2 0 4 に T C B ( b ) の領域を生成し ( ステップ S 5 0 1 ) 、優先度の領域に当該タスクの優先度を書き込み ( ステップ S 5 0 2 ) 、タスク情報の領域に当該タスクのタスク情報を書き込み ( ステップ S 5 0 3 ) 、 T C B リンクアドレスの領域に N U L L の値を代入する ( ステップ S 5 0 4 ) 。

## 【 0 0 6 5 】

ついで、タスク情報記憶部 2 0 4 に D C B ( b ) の領域を生成し ( ステップ S 5 0 5 ) 、相対 D L 値の領域に 0 の値を代入し ( ステップ S 5 0 6 ) 、最悪実行時間の領域に当該タスクの最悪実行時間を書き込み ( ステップ S 5 0 7 ) 、 D C B リンクアドレスの領域に D C B ( 1 ) の位置を示すアドレスを書き込み ( ステップ S 5 0 8 ) 、 T C B リンクアドレスの領域に T C B ( b ) の位置を示すアドレスを書き込む ( ステップ S 5 0 9 ) 。

## 【 0 0 6 6 】

更に、続いて、DCB (1) の相対DL値の領域に、(T-当該タスクのDL時刻) の値を上書きし (ステップS 5 1 0)、タスク情報記憶部 2 0 4 におけるDL基準時刻 3 0 1 の領域に当該タスクのDL時刻を上書きし (ステップS 5 1 1)、DCBリンクアドレス 3 0 2 の領域にDCB (b) の位置を示すアドレスを上書きする (ステップS 5 1 2)。

#### 【0 0 6 7】

この後、タイマ制御部 2 2 に当該タスクのDL時刻を出力する (ステップS 5 1 3)。

### ③ 処理Cの動作

処理Cの動作を、図9に示すフローチャートを用いて説明する。

スケジューリング部 2 0 3 は、タスク情報記憶部 2 0 4 にTCB (c) の領域を生成し (ステップS 6 0 1)、優先度の領域に当該タスクの優先度を書き込み (ステップS 6 0 2)、タスク情報の領域に当該タスクのタスク情報を書き込み (ステップS 6 0 3)、TCBリンクアドレスの領域にNULLの値を代入する (ステップS 6 0 4)。

#### 【0 0 6 8】

この後、タスク情報記憶部 2 0 4 にDCB (c) の領域を生成し (ステップS 6 0 5)、相対DL値の領域に (当該タスクのDL時刻-T) の値を代入し (ステップS 6 0 6)、最悪実行時間の領域に当該タスクの最悪実行時間を書き込み (ステップS 6 0 7)、DCBリンクアドレスの領域にNULLの値を代入し (ステップS 6 0 8)、TCBリンクアドレスの領域にTCB (c) の位置を示すアドレスを書き込む (ステップS 6 0 9)。

#### 【0 0 6 9】

つづいて、DCB (i) のDCBリンクアドレスの領域に、DCB (c) の位置を示すアドレスを上書きする (ステップS 6 1 0)。

### ④ 処理Dの動作

処理Dの動作を、図10に示すフローチャートを用いて説明する。

スケジューリング部 2 0 3 は、タスク情報記憶部 2 0 4 にTCB (d) の領域を生成し (ステップS 7 0 1)、優先度の領域に当該タスクの優先度を書き込み

(ステップS 7 0 2)、タスク情報の領域に当該タスクのタスク情報を書き込み  
(ステップS 7 0 3)、TCBリンクアドレスの領域にNULLの値を代入する  
(ステップS 7 0 4)。

#### 【0 0 7 0】

この後、タスク情報記憶部2 0 4にDCB (d) の領域を生成し(ステップS 7 0 5)、  
相対DL値の領域に(当該タスクのDL時刻-T)の値を代入し(ステップS 7 0 6)、  
最悪実行時間の領域に当該タスクの最悪実行時間を書き込み(ステップS 7 0 7)、  
DCB (d) のDCBリンクアドレスの領域にDCB (i) の位置を示すアドレスを書き込み  
(ステップS 7 0 8)、TCBリンクアドレスの領域にTCB (d) の位置を示すアドレスを書き込む  
(ステップS 7 0 9)。

#### 【0 0 7 1】

続いて、DCB (i - 1) のDCBリンクアドレスの領域に、DCB (d) の  
位置を示すアドレスを上書きする(ステップS 7 1 0)。

### ⑤ 処理Eの動作

処理Eの動作を、図1 1に示すフローチャートを用いて説明する。

スケジューリング部2 0 3は、変数jを有し、初期値を1に設定し(ステップ  
S 8 0 1)、タスク情報記憶部2 0 4におけるDCB (i) に含まれるTCBリ  
ンクアドレスにより指定されるTCB (j) を認識し(ステップS 8 0 2)、T  
CB (j) の優先度を読み出す(ステップS 8 0 3)。

#### 【0 0 7 2】

ついで、当該タスクの優先度値とTCB (j) の優先度値とを比較する(ステ  
ップS 8 0 4)。

比較の結果、当該タスクの優先度 $\leq$ TCB (j) の優先度である場合(ステッ  
プS 8 0 4で $\leq$ )、変数jに1を加算し(ステップS 8 0 5)、TCB (j) の  
有無を判定する(ステップS 8 0 6)。

#### 【0 0 7 3】

TCB (j) がある場合(ステップS 8 0 6でYES)、ステップS 8 0 4の  
動作に戻る。

他方、TCB (j) がない場合 (ステップ S 8 0 6 で NO)、又は当該タスクの優先度 > TCB (j) の優先度である場合 (ステップ S 8 0 4 で >)、タスク情報記憶部 2 0 4 に TCB (e) の領域を生成し (ステップ S 8 0 7)、優先度の領域に当該タスクの優先度を書き込み (ステップ S 8 0 8)、タスク情報の領域に当該タスクのタスク情報を書き込み (ステップ S 8 0 9)、TCB リンクアドレスの領域に TCB (j) の位置を示すアドレスを書き込む (ステップ S 8 1 0)。

## 【 0 0 7 4 】

この後、TCB (j - 1) の TCB リンクアドレスの領域に、TCB (e) の位置を示すアドレスを書き込み (ステップ S 8 1 1)、DCB (i) の最悪実行時間の領域より、時間 H を読み出し (ステップ S 8 1 2)、読出した時間 H と当該タスクの最悪実行時間とを加算した合計時間を上書きする (ステップ S 8 1 3)。

## (3) タスク切換えの動作

タスク切換えの動作を、図 1 2 に示すフローチャートを用いて説明する。

## 【 0 0 7 5 】

タスク切換え部 2 0 5 は、受取った信号の判別を行う (ステップ S 9 0 1)。

タイムアウト信号を受け取った場合 (ステップ S 9 0 1 でタイムアウト信号)、処理 F を行い、終了信号を受け取った場合 (ステップ S 9 0 1 で終了信号)、TCB 4 1 に含まれる TCB リンクアドレス 4 1 3 を読み出し (ステップ S 9 0 2)、読出した TCB リンクアドレス 4 1 3 の値が NULL 値であるか否かを判定する (ステップ S 9 0 3)。

## 【 0 0 7 6 】

判定の結果、前記 TCB リンクアドレス 4 1 3 の値が、NULL 値である場合 (ステップ S 9 0 3 で YES)、処理 F を行い、他方、NULL 値でない場合 (ステップ S 9 0 3 で NO)、処理 G を行う。

## ① 処理 F の動作

処理 F の動作を、図 1 3 に示すフローチャートを用いて説明する。

## 【 0 0 7 7 】

タスク切換部 2 0 5 は、タスク情報記憶部 2 0 4 より先頭ポインタ、DCB 4 0 及び DCB 5 0 を読み出し（ステップ S 9 1 1）、読出した先頭ポインタの DL 基準時刻の領域に DL 基準時刻と DCB 5 0 に含まれる相対 DL 値とを合計した時刻を上書きする（ステップ S 9 1 2）。

続いて、先頭ポインタの DCB リンクアドレスの領域に DCB 5 0 の位置を示すアドレスを上書きし（ステップ S 9 1 3）、DCB 5 0 の相対 DL 値の領域に 0 を上書きする（ステップ S 9 1 4）。

#### 【 0 0 7 8 】

この後、DCB 4 0 及び DCB 4 0 に接続される全ての TCB をタスク情報記憶部 2 0 4 より消去し（ステップ S 9 1 5）、DCB 5 0 の TCB リンクアドレスにより指定される TCB 5 1 を読み出し、読出した TCB 5 1 に含まれるタスク情報を実行制御部 2 3 へ出力する（ステップ S 9 1 6）。

#### ② 処理 G の動作

処理 G の動作を、図 1 4 に示すフローチャートを用いて説明する。

#### 【 0 0 7 9 】

タスク切換部 2 0 5 は、タスク情報記憶部 2 0 4 より DCB 4 0、TCB 4 1 及び TCB 4 2 を読み出し（ステップ S 9 2 1）、DCB 4 0 に含まれる TCB リンクアドレスの領域に、TCB 4 2 の位置を示すアドレスを上書きする（ステップ S 9 2 2）。

ついで、TCB 4 1 をタスク情報記憶部 2 0 4 より消去し（ステップ S 9 2 3）、TCB 4 2 に含まれるタスク情報を実行制御部 2 3 へ出力する（ステップ S 9 2 4）。

#### [ 4 ] その他の変形例

本発明の趣旨を逸脱しない範囲で、上記の実施の形態以外の形態で実施することができる。以下に、その例を示す。

（１） タスクに割り当てられる優先度は、本実施の形態における DL 時刻と優先度との 2 種類の優先度に限定されない。複数の種類の優先度を適用してもよい。なお、複数の種類の優先度には、階層関係が与えられているものとする。

#### 【 0 0 8 0 】



また、タスクに割り当てられる優先度は、DL時刻の1種類としてもよい。以下に、DL時刻のみを割り当てられたタスクのスケジューリング方法について、実施の形態と異なる内容を説明する。

タスク情報記憶部204は、各タスクに割り当てられたDL時刻のうち所定の単位時間の整数倍からなる第1優先度と、前記DL時刻のうち前記所定の単位時間未満の部分からなる第2優先度との2種類の優先度を管理するために設けられており、図3に示すように、先頭ポインタ30、グループ400、グループ500、・・・を有している。例えば、DL時刻が“時/分/秒/100ミリ秒”で構成される場合、前記第1優先度は“時/分/秒”単位部分からなり、前記第2優先度は“100ミリ秒”単位部分からなる。

#### 【0081】

先頭ポインタ30、グループ400、グループ500、・・・は、先頭ポインタ30を先頭とし、グループ400、グループ500、・・・の順に配列を構成している。

グループ400、グループ500、・・・は、第1優先度に基づいて並べられる。

#### 【0082】

グループ400は、同一の第1優先度が割り当てられた複数のタスクに関する情報をまとめて管理するためのものであり、DCB40、TCB41、TCB42、・・・からなる。

DCB40、TCB41、TCB42、・・・は、DCB40を先頭とし、TCB41、TCB42、・・・の順に配列を構成している。TCB41、TCB42、・・・は、第2優先度に基づいて並べられる。

#### 【0083】

スケジューリング部203は、当該タスクを管理するTCBを、第1優先度と第2優先度とに基づいて、記憶部204において定まるメモリ空間上の位置に書き込みを行う。

(2) DL時刻のデータ構造は、本実施の形態における“時/分/秒”に限定されない。その他の妥当な単位時間により構成されるときもよい。

(3) 優先度に用いる値は、本実施の形態における“高”、“中”及び“下”の3レベルに限定されない。その他の妥当な基準値を適用してもよい。

(4) 本実施の形態において、DL時刻の最低値を示すデータ構造として“FF/FF/FF”を用いたが、その他の妥当なデータ値を設定しても良い。

(5) 本発明は、上記の実施の形態におけるタスク切換部205が、プリエンプトと呼ばれる排他制御を行うとしてもよい。

【0084】

以下に、タスク切換部205が行う排他制御について説明する。

タスク切換部205は、タイムアウト信号又は終了信号を受取った場合、実行していたタスクを識別するTCBのTCBリンクアドレスの領域に格納されたデータをタスク情報記憶部204より読み出す。

タスク切換部205は、読出したデータにより指定されるTCBのタスク情報の領域にロック情報が有るか否かの判定を行う。ロック情報とは、当該TCBにより識別されるタスクが排他制御により実行出来ない状態にあるか否かの制御情報である。

【0085】

タスク切換部205は、タスク情報の領域にロック情報が有る場合、さらにTCBリンクアドレスの領域に格納されたデータを読み出し、前記判定を実行する。タスク切換部205は、前記読み出し、及び前記判定をタスク情報の領域にロック情報が有る限り、繰り返し行う。

タスク切換部205は、タスク情報の領域にロック情報が無い場合、本実施の形態と同様なタスク切換え処理を実行する。

(6) 本発明は、上記の実施の形態に示す方法であるとしてもよい。また、これらの方法をコンピュータにより実現するためのコンピュータプログラムであるとしてもよいし、前記コンピュータプログラムからなるデジタル信号であるとしてもよい。

【0086】

また、本発明は、前記コンピュータプログラム又は前記デジタル信号をコンピュータ読み取り可能な記録媒体、例えば、ROM、RAM、フレキシブルディスク

ク、ハードディスク、CD-ROM、MO、DVD、DVD-ROM、DVD-RAM、半導体メモリなど、に記録したものとしてもよい。また、これらの記録媒体に記録されている前記コンピュータプログラム又は前記デジタル信号であるとしてもよい。

【 0 0 8 7 】

また、本発明は、前記コンピュータプログラム又は前記デジタル信号を、電気通信回線、無線又は有線通信回路、インターネットを代表とするネットワーク等を経由して伝送するものとしてもよい。

また、本発明は、マイクロプロセッサとメモリとを備えたコンピュータシステムであって、前記メモリは、上記コンピュータプログラムを記憶しており、前記マイクロプロセッサは、前記コンピュータプログラムに従って動作するとしてもよい。なお、前記コンピュータシステムには、TV、冷蔵庫、電話機、エアコンなどの家電製品も含まれる。

【 0 0 8 8 】

また、前記プログラム又は前記デジタル信号を前記記録媒体に記録して移送することにより、又は前記プログラム又は前記デジタル信号を前記ネットワーク等を経由して移送することにより、独立した他のコンピュータシステムにより実施するとしてもよい。

(7) 上記実施の形態及び上記変形例をそれぞれ組み合わせるとしてもよい。

【 0 0 8 9 】

【発明の効果】

以上説明したように、本発明は、それぞれ定められた完了目標時刻以前に完了すべき複数のタスクの実行順序を決定し、各タスクを実行するプログラム実行装置であって、各タスクには、階層関係を有する複数の種類の優先度が割り当てられており、すでに生成されているタスク毎に、当該タスクに割り当てられた複数の種類の優先度に基づいて定まるメモリ空間上の位置において当該タスクを識別する識別子を記憶している記憶手段と、新たに生成されたタスクに割り当てられた複数の種類の優先度、及び当該タスクを識別する識別子を受け付ける受付手段と、前記記憶手段において、前記受付手段により受付けた複数の種類の優先度

に基づいて定まるメモリ空間上の位置に前記受付けた識別子を書き込む書込手段とを備え、複数の種類の優先度に基づいて前記記憶手段において各識別子が配列されている位置の順序に従って、各識別子が識別するタスクの実行順序を決定することを特徴とする。

#### 【 0 0 9 0 】

この構成によると、プログラム実行装置は、タスクの実行順序を、階層関係を有する複数の種類の優先度により適切に制御することが可能となる。

ここで、各タスクには、前記複数の種類の優先度として、当該タスクの完了目標時刻を示す第 1 優先度と、前記第 1 優先度より下位の優先度である第 2 優先度とが割り当てられており、前記記憶手段は、第 1 優先度として同じ完了目標時刻が割り当てられたタスクが同じグループに属するように、前記複数のタスクを分類して生成した複数のグループを有し、各グループは完了目標時刻の順に従って定まるメモリ空間上の位置に配列されており、各グループにはタスクを識別する識別子が当該タスクに割り当てられた第 2 優先度に従って定まるメモリ空間上の位置に配列されており、前記受付手段は、新たに生成されたタスクに割り当てられた第 1 優先度としての完了目標時刻、当該タスクに割り当てられた第 2 優先度、及び当該タスクを識別する識別子を受け付け、前記書込手段は、前記受付けた第 1 優先度と同じ優先度を持つ識別子が含まれるグループにおいて、前記受付けた第 2 優先度に基づいて定まるメモリ空間上の位置に前記受付けた識別子を書き込み、前記プログラム実行装置は、前記記憶手段において、第 1 優先度及び第 2 優先度に基づいて各識別子が配列されている位置の順序に従って、各識別子が識別するタスクの実行順序を決定する。

#### 【 0 0 9 1 】

この構成によると、プログラム実行装置は、同一な完了目標時刻を割り振られた複数のタスクの実行順序を、第 2 優先度により適切に制御することが可能となる。

ここで、前記記憶手段は、直近の完了目標時刻である第 1 優先度に基づいてタスクが分類されたグループである先頭グループに対応付けて、前記完了目標時刻である第 1 優先度を記憶しており、前記先頭グループに後続して配列されたグル

ープである後続グループ毎に、当該後続グループに含まれるタスクに割り当てられた完了目標時刻と、当該後続グループに先行して配列されているグループに含まれるタスクに割り当てられた完了目標時刻との時間差を、当該後続グループに対応づけて記憶しており、前記書込手段は、前記受付手段により受付けた完了目標時刻と、前記直近の完了目標時刻とを比較して、一致する場合、前記先頭グループにおいて前記受付けた第2優先度に基づいて定まるメモリ空間上の位置に前記受付けた識別子を書き込み、一致しない場合、前記後続グループ毎に、当該後続グループに対応づけられた時間差と、当該後続グループに先行して配列されているグループに含まれるタスクに割り当てられた完了目標時刻とを加算して得られる時刻と前記受付手段により受付けた完了目標時刻とを比較し、一致する場合に、当該後続グループにおいて前記受付けた第2優先度に基づいて定まるメモリ空間上の位置に前記受付けた識別子を書き込む。

## 【0092】

この構成によると、記憶手段は各後続グループに対応づけて前記時間差を記憶していることにより、書込手段が前記書込み処理において必要とするデータ量の削減を可能とする。

ここで、第1優先度が取り得る値のうちの最も低い値は、完了目標時刻を構成する所定ビット列が取り得る最大値であり、前記記憶手段は、前記所定ビット列の最大値を有する完了目標時刻が割り当てられたタスクを含むグループを、前記完了目標時刻の順に従って定まる最後尾の位置において記憶しており、前記受付手段は、新たに生成されたタスクに割り当てられた前記所定ビット列の最大値を有する完了目標時刻、当該タスクに割り当てられた第2優先度、及び当該タスクを識別する識別子を受け付け、前記書込手段は、前記最後尾に位置するグループにおいて前記受付けた第2優先度に基づいて定まるメモリ空間上の位置に前記受付けた識別子を書き込む。

## 【0093】

この構成によると、プログラム実行装置は、タスクの第1優先度を最も低い値に制御することが可能となる。

ここで、前記記憶手段は、さらに、各グループについて、当該グループに含ま

れる全てのタスクの実行に要すると予測される総実行時間を、当該グループに対応づけて記憶しており、前記受付手段は、新たに生成されたタスクの実行に要すると予測される実行時間を受け付け、前記プログラム実行装置は、さらに、現在時刻に、前記総実行時間と、前記実行時間とを加算して得られた時刻が、前記受付けた識別子により識別されるタスクに割り当てられた完了目標時刻より前であるか否かを判定し、前記時刻は前記完了目標時刻以降の時刻である場合に、前記受付けた識別子により識別されるタスクの実行を拒否する拒否信号を出力する判定手段を備え、前記判定手段により前記時刻は前記完了目標時刻より前の時刻であると判定される場合、前記書込手段は、前記受付けた第 1 優先度と同じ優先度を持つ識別子が含まれるグループにおいて、前記受付けた第 2 優先度に基づいて定まるメモリ空間上の位置に前記受付けた識別子を書き込む。

## 【 0 0 9 4 】

この構成によると、プログラム実行装置は、タスクの実行を完了目標時刻以前に完了させるように制御することが可能となる。

ここで、前記判定手段は、前記グループに後続する全ての後続グループから当該後続グループを選択し、選択した当該後続グループについて、現在時刻に、前記当該後続グループに対応付けられた総実行時間と、前記受付けた実行時間とを加算して得られた時刻が、前記当該後続グループに含まれるタスクに割り当てられた完了目標時刻より前であるか否かを判定し、前記の選択と判定とを全ての後続グループについて繰り返し、前記判定手段による全ての判定において前記時刻は前記完了目標時刻より前であると判定される場合、前記書込手段は、前記受付けた第 1 優先度と同じ優先度を持つ識別子が含まれるグループにおいて、前記受付けた第 2 優先度に基づいて定まるメモリ空間上の位置に前記受付けた識別子を書き込み、前記判定手段による全ての判定のうち少なくとも 1 回前記時刻は前記完了目標時刻以降の時刻である判定される場合に、前記判定手段は前記受付けた識別子により識別されるタスクの実行を拒否する拒否信号を出力する。

## 【 0 0 9 5 】

この構成によると、プログラム実行装置は、さらに、記憶手段による記憶されている全てのタスクの実行を完了目標時刻以前に完了させるように制御すること

が可能となる。

ここで、各タスクには、当該タスクに定められた完了目標時刻のうち所定の単位時間の整数倍からなる第 1 優先度と、前記完了目標時刻のうち前記所定の単位時間未満の部分からなる第 2 優先度とが割り当てられており、前記記憶手段は、同じ第 1 優先度が割り当てられたタスクが同じグループに属するように、前記複数のタスクを分類して生成した複数のグループを有し、各グループは第 1 優先度に従って定まるメモリ空間上の位置に配列されており、各グループにはタスクを識別する識別子が当該タスクに割り当てられた第 2 優先度に従って定まるメモリ空間上の位置に配列されており、前記受付手段は、新たに生成されたタスクに割り当てられた第 1 優先度、第 2 優先度、及び当該タスクを識別する識別子を受け付け、前記書込手段は、前記受付けた第 1 優先度と同じ優先度を持つ識別子が含まれるグループにおいて、前記受付けた第 2 優先度に基づいて定まるメモリ空間上の位置に前記受付けた識別子を書き込み、前記プログラム実行装置は、前記記憶手段において、第 1 優先度及び第 2 優先度に基づいて各識別子が配列されている位置の順序に従って、各識別子が識別するタスクの実行順序を決定する。

【 0 0 9 6 】

この構成によると、タスクに割り当てられる優先度が完了目標時刻の 1 種類である場合に、プログラム実行装置は、タスクの実行順序を適切に制御できる。

【図面の簡単な説明】

【図 1】 携帯電話機 1 の構成を示すブロック図である。

【図 2】 タスクスケジューリング部 2 0 の機能を示すブロック図である。

【図 3】 タスク情報記憶部 2 0 4 におけるスケジューリング状況を示すブロック図である。

【図 4】 タスクスケジューリング部 2 0 によるタスクスケジューリングの動作を示すシーケンス図である。

【図 5】 スケジューリング可能性判定の動作を示すフローチャートである。

【図 6】 スケジューリング動作を示すフローチャートである。

【図 7】 処理 A の動作を示すフローチャートである。

【図 8】 処理 B の動作を示すフローチャートである。

【図 9】 処理 C の動作を示すフローチャートである。

【図 1 0】 処理 D の動作を示すフローチャートである。

【図 1 1】 処理 E の動作を示すフローチャートである。

【図 1 2】 タスク切換え動作を示すフローチャートである。

【図 1 3】 処理 F の動作を示すフローチャートである。

【図 1 4】 処理 G の動作を示すフローチャートである。

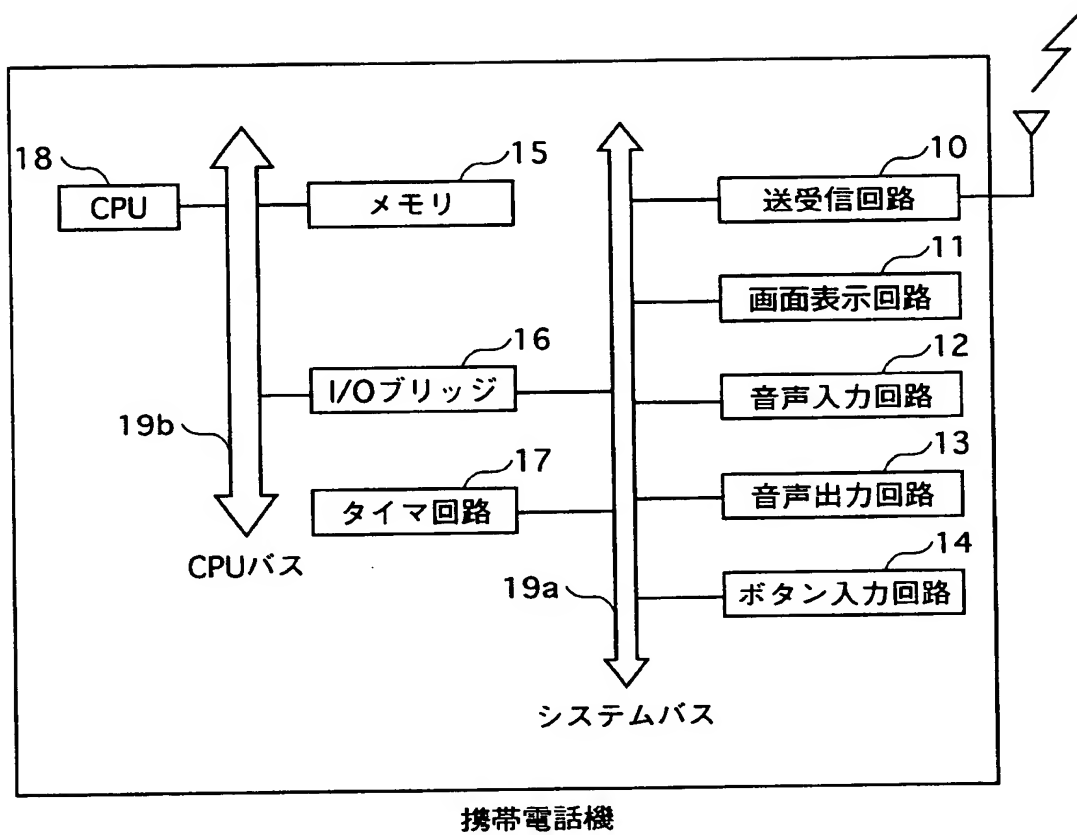
【符号の説明】

1	携帯電話機
1 0	送受信回路
1 1	画面表示回路
1 2	音声入力回路
1 3	音声出力回路
1 4	ボタン入力回路
1 5	メモリ
1 6	I/Oブリッジ
1 7	タイマ回路
1 8	CPU
2 0	タスクスケジューリング部
2 1	プログラム記憶部
2 2	タイマ制御部
2 3	実行制御部
2 0 1	タスク受付部
2 0 2	スケジューリング可能性判定部
2 0 3	スケジューリング部
2 0 4	タスク情報記憶部
2 0 5	タスク切換部



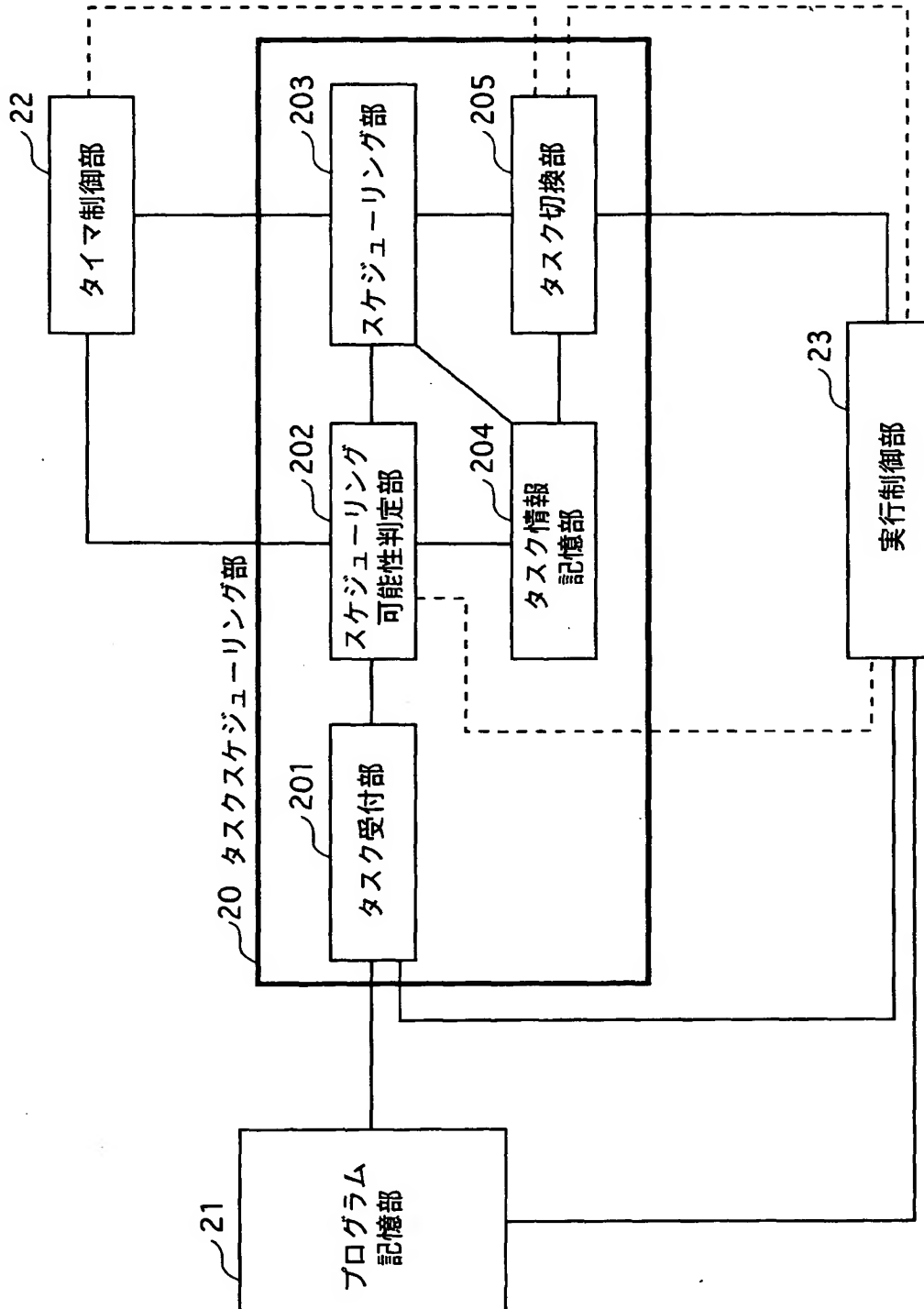
【書類名】 図面

【図 1】

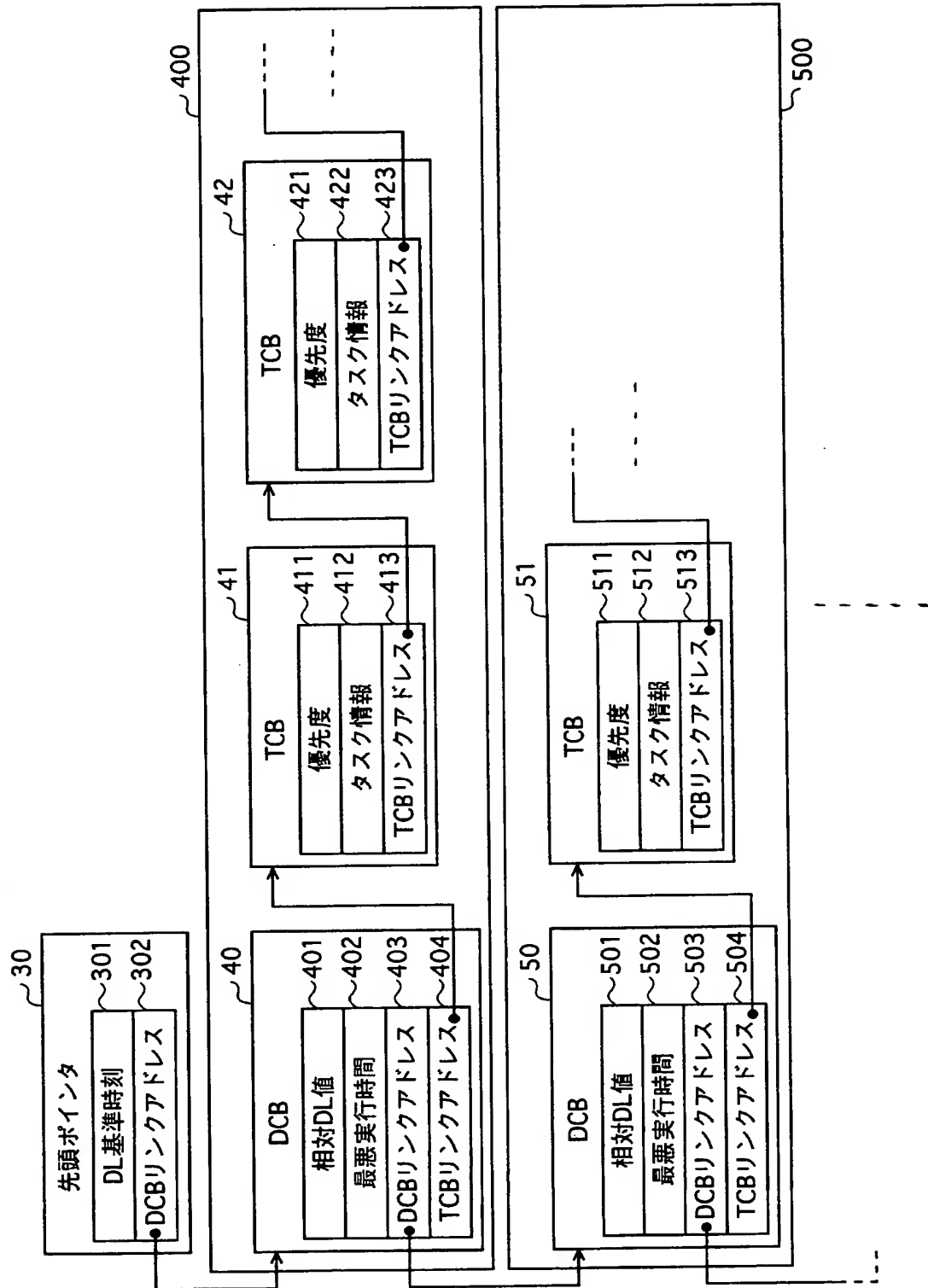


1

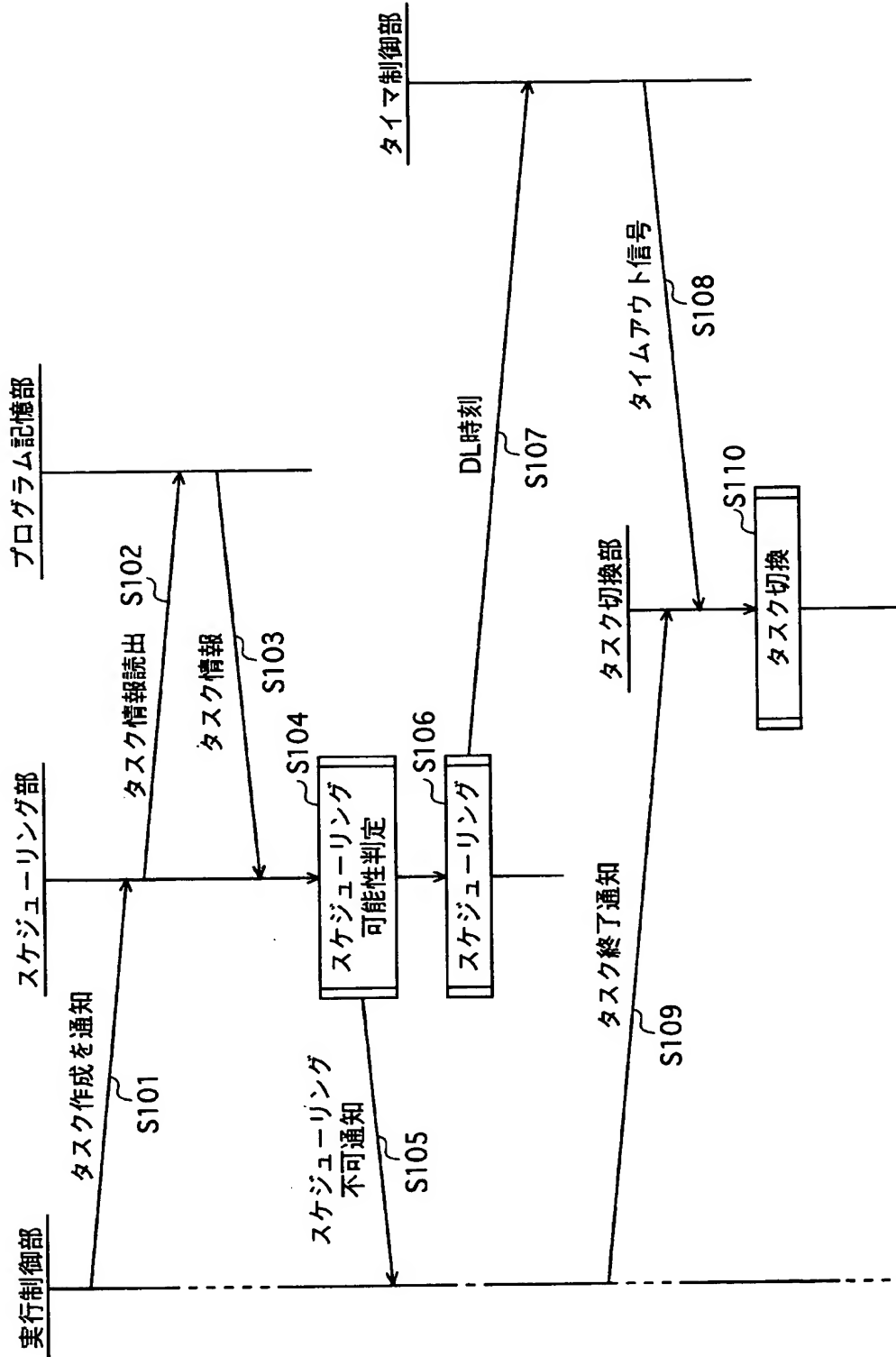
【図 2】



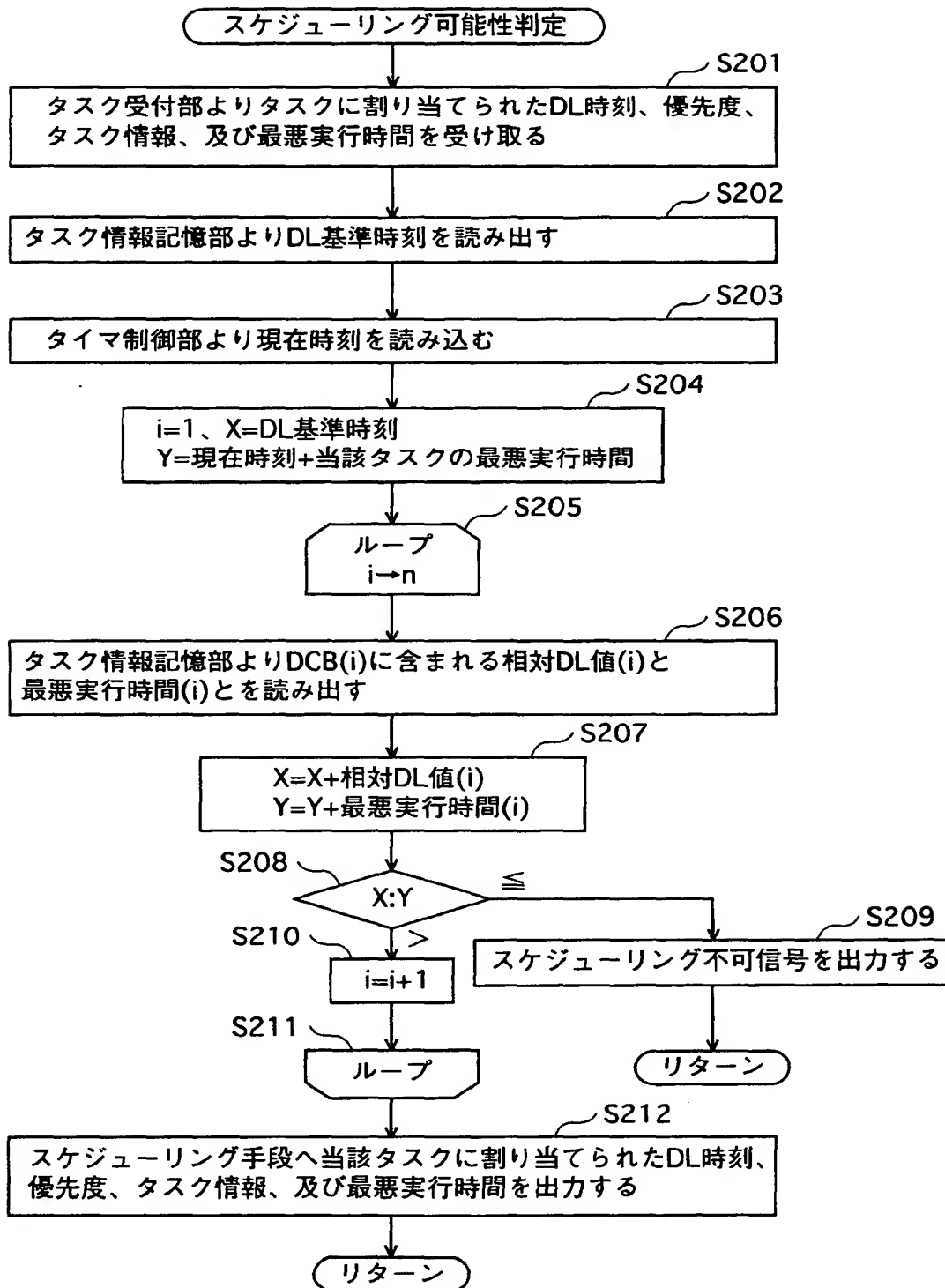
【図3】



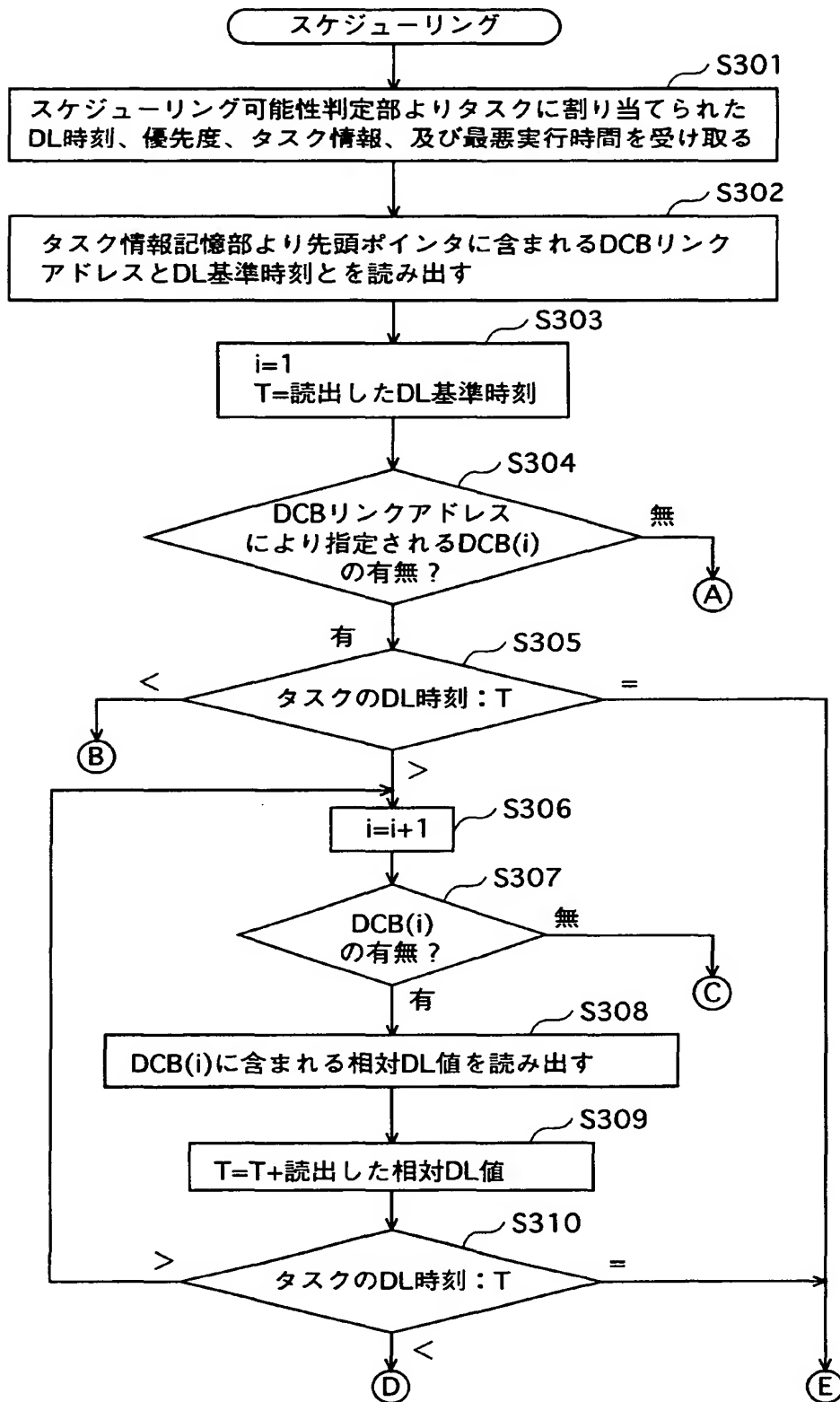
【図4】



【図 5】



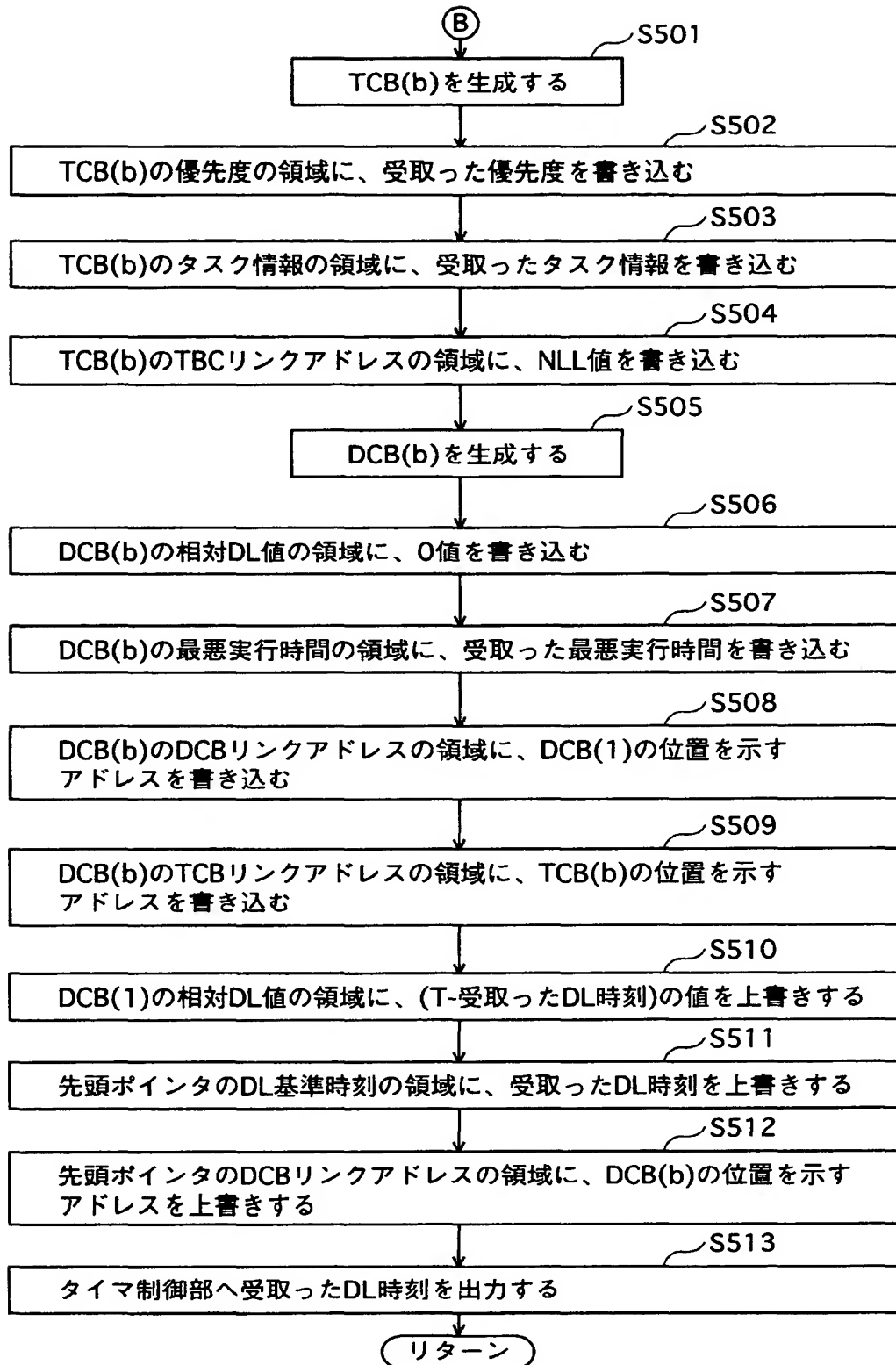
【図 6】



【図 7】

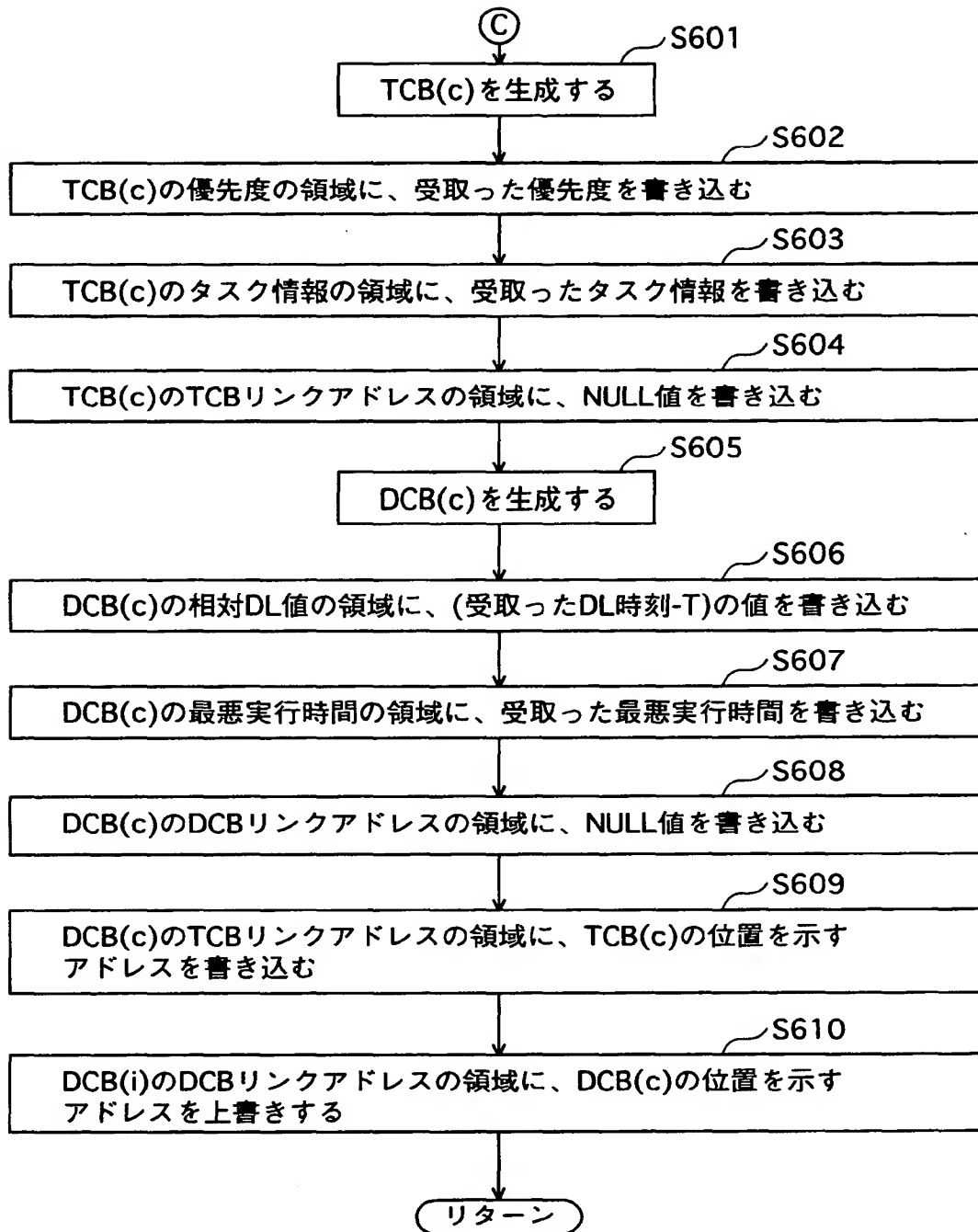


【図 8】

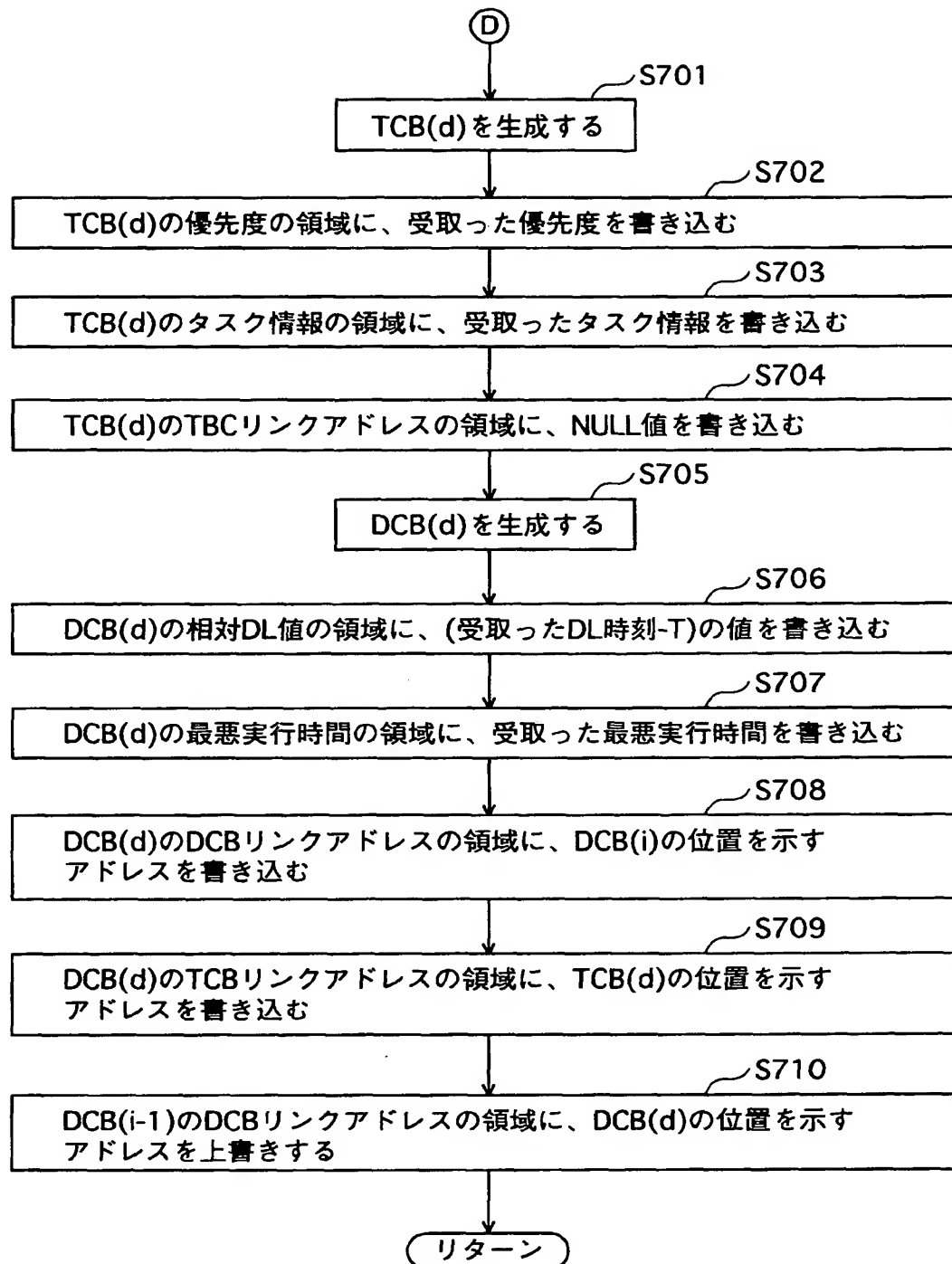




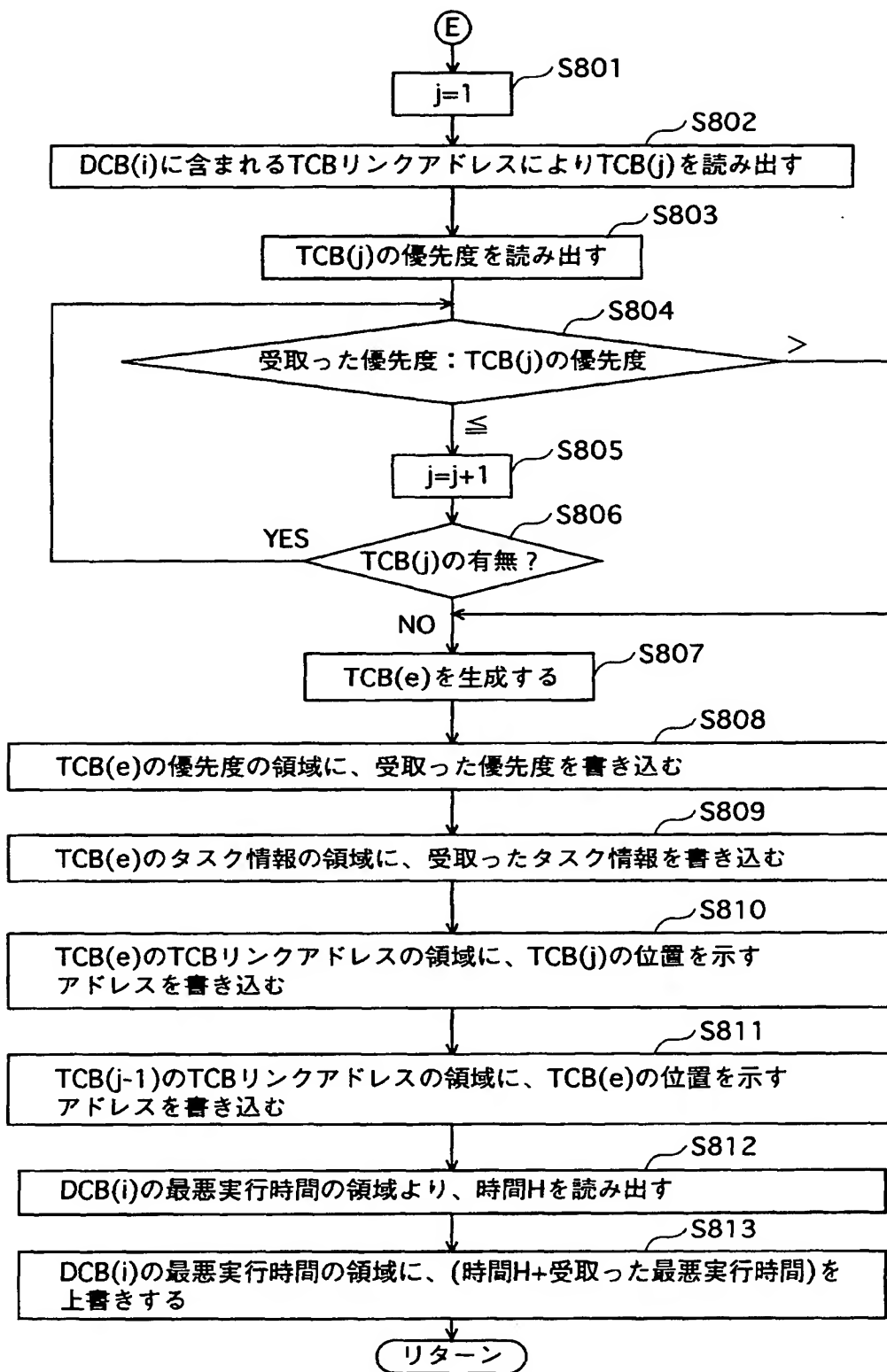
【図 9】



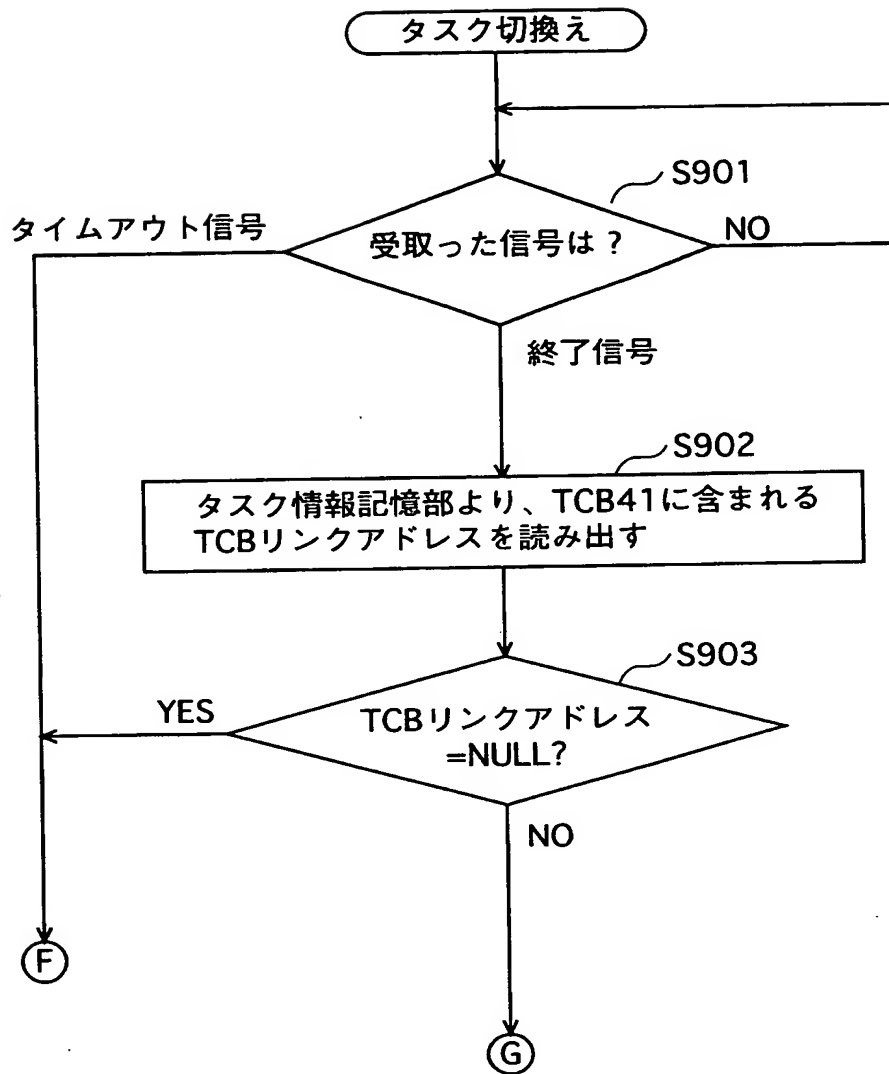
【図 1 0】



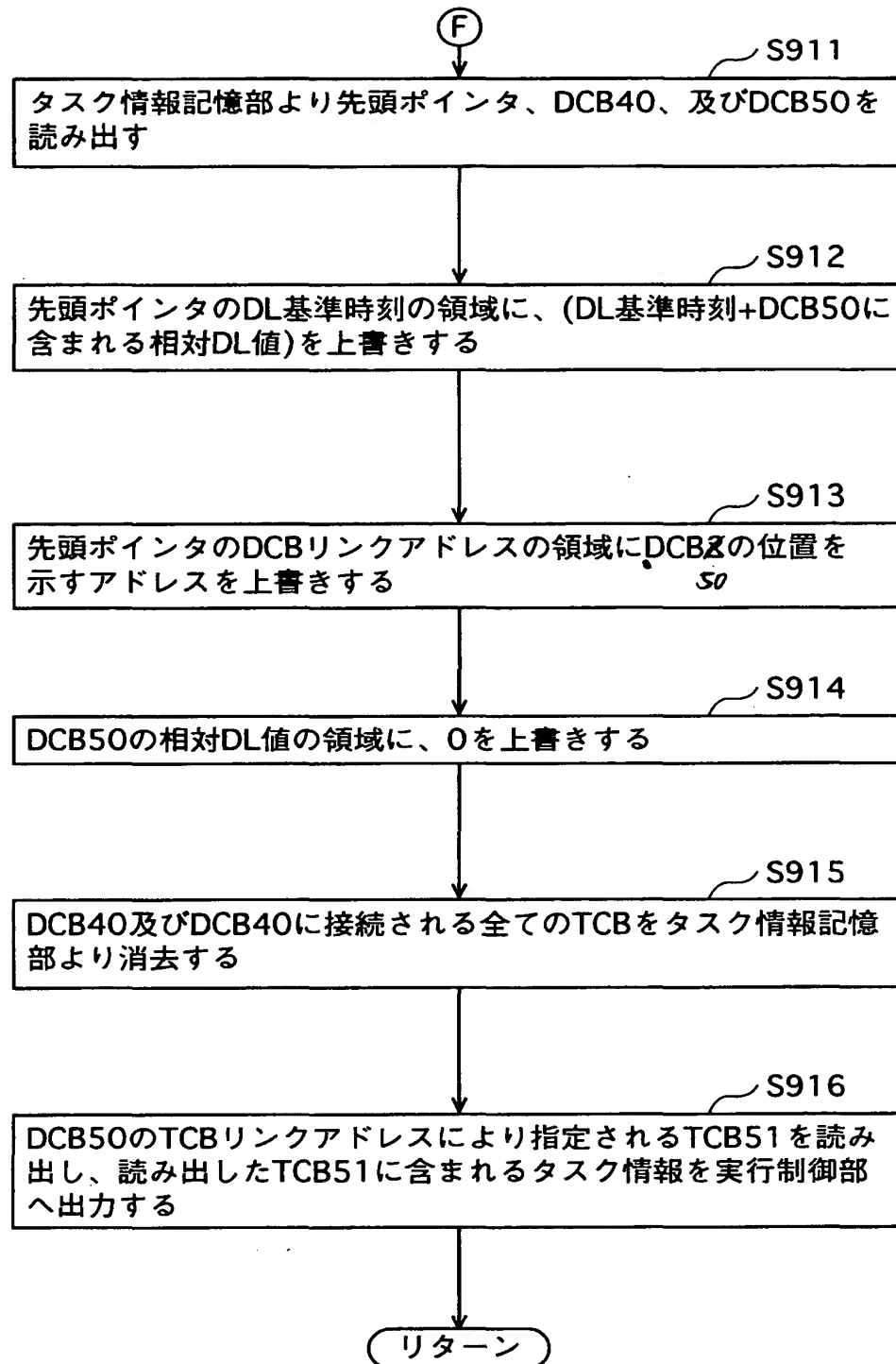
【図11】



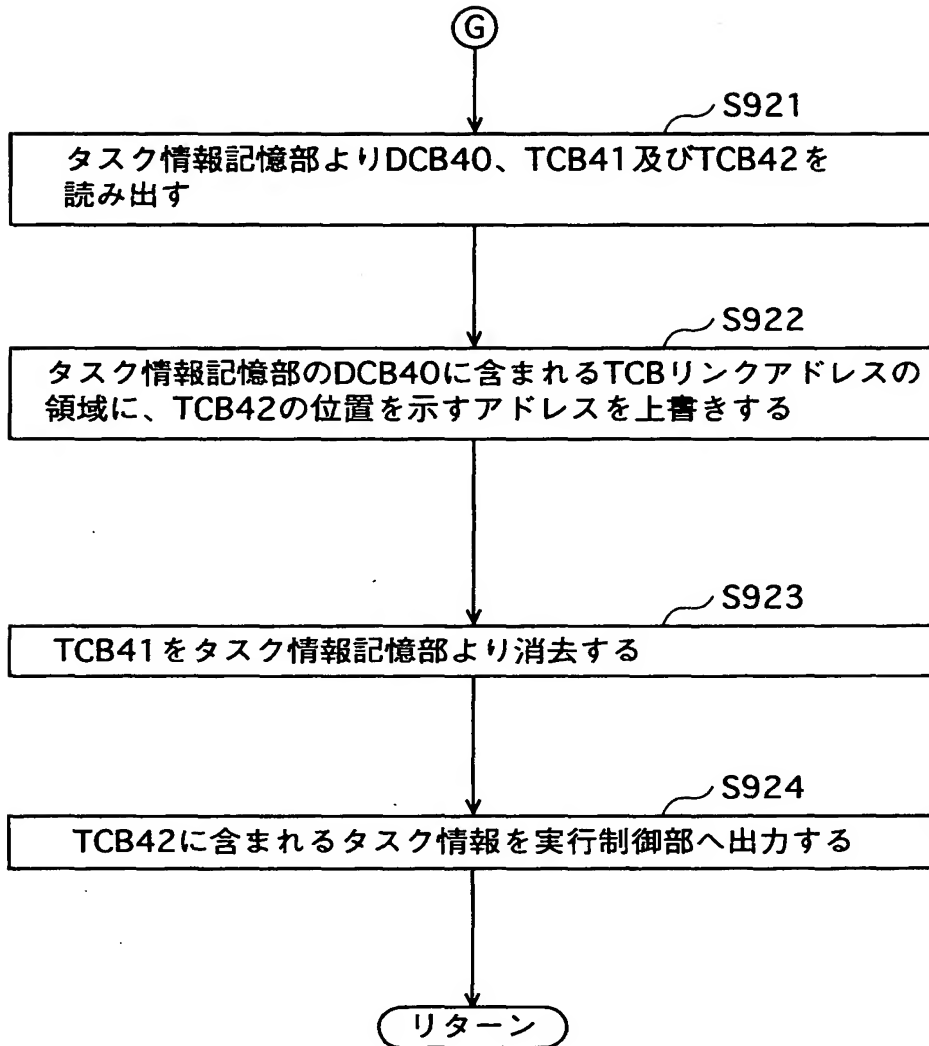
【図 1 2】



【図 1 3】



【図 1 4】



【書類名】 要約書

【要約】

【課題】 タスクの実行順序を適切に決定し、決定した実行順序に従ってタスクを実行するプログラム実行装置を提供すること。

【解決手段】 定められた完了目標時刻以前に完了すべき複数のタスクの実行順序を決定し、各タスクを実行するプログラム実行装置であって、各タスクには、階層関係を有する複数の種類の優先度が割り当てられ、生成されているタスク毎に、当該タスクに割り当てられた複数の種類の優先度に基づいて定まるメモリ空間上の位置において当該タスクを識別する識別子を記憶している記憶手段と、新たに生成されたタスクに割り当てられた複数の種類の優先度、及び当該タスクを識別する識別子を受け付ける受付手段と、前記記憶手段において、前記受け付けた複数の種類の優先度に基づいて定まるメモリ空間上の位置に前記受け付けた識別子を書き込む書込手段とを備えることを特徴とする。

【選択図】 図 3

出 願 人 履 歴 情 報

識別番号 [ 0 0 0 0 0 5 8 2 1 ]

1. 変更年月日	1 9 9 0 年 8 月 2 8 日
[ 変更理由 ]	新規登録
住 所	大阪府門真市大字門真 1 0 0 6 番地
氏 名	松下電器産業株式会社